Modeling the Human Visuo-Motor System for Remote-Control Operation

A DISSERTATION
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Jonathan Andersh

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
Doctor of Philosophy

Advisors: Bérénice Mettler and Nikolaos Papanikolopoulo

May, 2018

Acknowledgements

I am grateful to all the people who have provided invaluable support, guidance, and assistance during my time at University of Minnesota.

First, I would like to thank my thesis advisor, Prof. Bérénice Mettler. She provided guidance, encouragement, and support over the course of my thesis work. The frequent discussions we had on new ideas and different ways to approach research problems helped keep the work moving forward. I would also like to thank my co-advisor Nikolaos Papanikolopoulos for sharing his experience and providing support.

I want to express my appreciation to my other dissertation committee members, Prof. Volkan Isler and Dr. Vassilios Morellas, for their availability and their time.

I would like to thank my colleagues and friends Navid Dadkhah, Zhaodan Kong, and Bin Li, at the Interactive Guidance and Control Lab, and Anoop Cherian from the Center for Distributed Robotics. They made working in the lab enjoyable. I could not ask for a better set of colleagues.

Special thanks to the administrative staff of the Department of Conputer Science and Engineering, and the Graduate School.

Last, I would like to thank my mother to whom I am eternally grateful. She has been a constant source of inspiration, stability, and encouragement.

Dedication

To my mother for her support, love, and encouragement.

Abstract

Successful operation of a teleoperated miniature rotorcraft relies on capabilities including guidance, trajectory following, feedback control, and environmental perception. For many operating scenarios fragile automation systems are unable to provide adequate performance. In contrast, human-in-the-loop systems demonstrate an ability to adapt to changing and complex environments, stability in control response, high level goal selection and planning, and the ability to perceive and process large amounts of information. Modeling the perceptual processes of the human operator provides the foundation necessary for a systems based approach to the design of control and display systems used by remotely operated vehicles. In this work we consider flight tasks for remotely controlled miniature rotorcraft operating in indoor environments. Operation of agile robotic systems in three dimensional spaces requires a detailed understanding of the perceptual aspects of the problem as well as knowledge of the task and models of the operator response. When modeling the human-in-the-loop the dynamics of the vehicle, environment, and human perception-action are tightly coupled in space and time. The dynamic response of the overall system emerges from the interplay of perception and action. The main questions to be answered in this work are: i) what approach does the human operator implement when generating a control and guidance response? ii) how is information about the vehicle and environment extracted by the human? iii) can the gaze patterns of the pilot be decoded to provide information for estimation and control? In relation to existing research this work differs by focusing on fast acting dynamic systems in multiple dimensions and investigating how the gaze can be exploited to provide action-relevant information. To study human-in-the-loop systems the development and integration of the experimental infrastructure is described. Utilizing the infrastructure, a theoretical framework for computational modeling of the human pilots perception-action is proposed and verified experimentally. The benefits of the human visuomotor model are demonstrated through application examples where the perceptual and control functions of a teleoperation system are augmented to reduce workload and provide a more natural human-machine interface.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Recent years have seen rapid advances in fields such as robotics and sensor technology that are fundamentally changing the way humans interact with the world. Improved robotics technology has led to an expanding number of applications that range from self driving cars [7, 8], robotic assisted surgery [9], and the wide availability of small scale unmanned aerial vehicles [10]. At the same time, sensor capabilities have advanced and can provide inexpensive measurements of human gaze and body motion. Combining these technologies allows for the investigation of human performance while conducting tasks requiring human-machine interaction. Fig. 1.1 shows a robotic UAV operating in the Interactive Guidance and Control (IGCL) lab where the operator gaze and motion can be measured. A systematic modeling approach utilizes the data captured from experimental flight tests to characterize the human pilot's interaction with the vehicle and environment. The primary objective of the work that follows is to utilize the pilot's gaze to model the human perception-action processes and implement augmentations for a teleoperation configuration.

The work in this dissertation is based around the multi-loop model of human control shown in Fig. 1.2. The model is defined as a hierarchical control system. For each level in the hierarchy, a reference is received from the next higher level. In addition, the higher level also chooses the appropriate perceptual processes as well as decision models to provide the necessary functionality (mental models in long term memory [11]). The models are switched when a higher level in the hierarchy determines that the operation of the lower level requires a different type of response. Based on the desired reference, the perceptual process extracts information from the visual stream to characterize the desired state as well as the actual state of the vehicle.

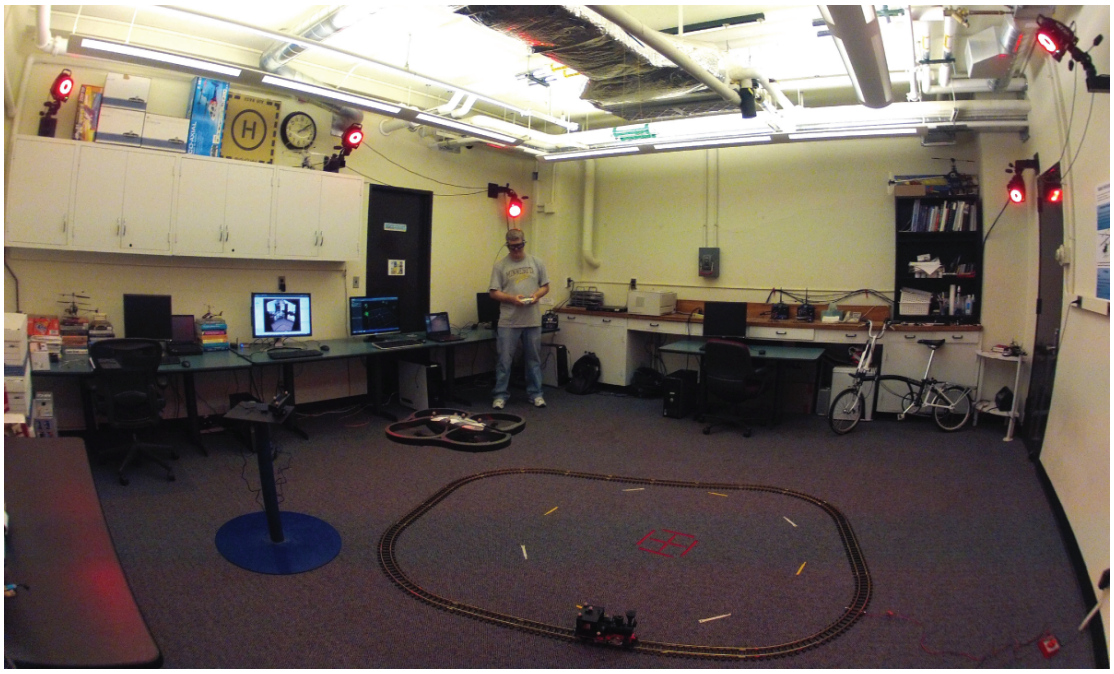


Figure 1.1: Lab facility for investigating human perception-action using a miniature UAV with motion and gaze sensing.

In Fig. 1.2 the outer loop performs a planning function that identifies the current subgoal and switches to the next subgoal when necessary. The next loop in the hierarchy, the perceptual guidance loop, generates trajectories that will maneuver the vehicle to the defined subgoal. The inner loop performs tracking and pursuit that follows the desired trajectory while rejecting disturbances. To perform these functions perceptual processes are required to extract information about the vehicle state, local subgoal, and environmental affordances. The different blocks in the diagram represent the system components that compromise a human pilot's perception and action.

1.1 Motivation

Dynamic teleoperation in complex environments requires a human to extract task relevant information from the optic array and generate an appropriate control response. Successful teleoperation relies on capabilities including guidance, trajectory following, feedback control, and

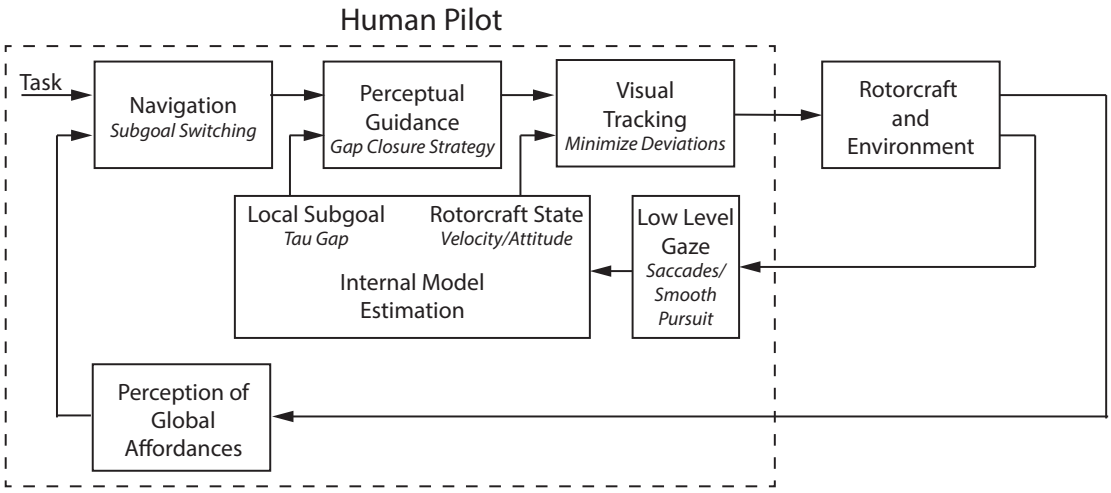


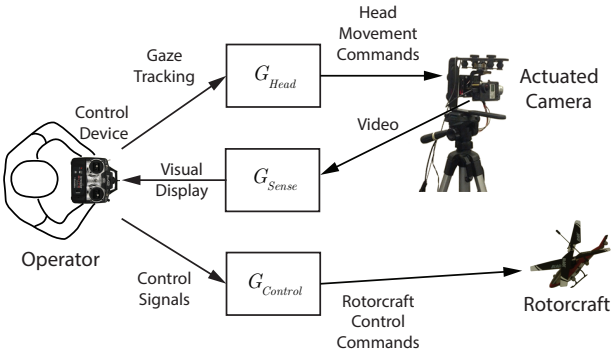
Figure 1.2: General framework for human perception and action.

environmental perception. Modeling the perceptual and control processes of the human operator provides the foundation necessary for a systems based approach to the design of control and user displays used by remotely operated vehicles. This work considers flight tasks for remotely controlled miniature rotorcraft operating in indoor environments. Operation of agile robotic systems in three dimensional spaces requires a detailed understanding of the perceptual aspects of the problem as well as knowledge of the task and models of the operator response. When modeling the human-in-the-loop the dynamics of the vehicle, environment, and human perception-action are tightly coupled. The dynamic response of the overall system emerges from the interplay of perception and action. The primary goal of this work is to investigate the structure of human pilot's perceptual and control processes and identify suitable models. Once identified, models of the human-in-the-loop can be used to design more natural and intuitive control interfaces that tap into the innate mechanisms and therefore reduce the operator workload and allow the human and automated systems to each act in their areas of strength.

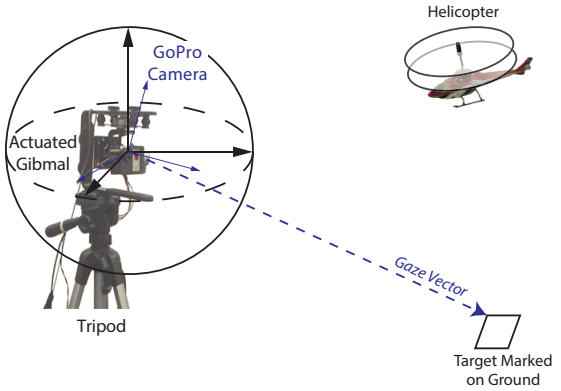
1.1.1 Teleoperation Example Application

Teleoperation applications cover a broad range of domains such as exploration, surgery, inspection, search and rescue, and surveillance. Remote teleoperation applications require the operator to perform tasks based on limited information using perceptual processes that are

usually structured to conform to the hardware constraints rather than natural human capabilities. Typical hardware constraints include inadequate video resolution, limited field of view, and poor depth information. Successful completion of remote tasks requires overcoming these limitations while maintaining situational awareness. This work investigates automating remote camera movement based on models of head-eye coordinated motion and augmenting the control system to assist the operator. The goal is to improve situation awareness and reduce operator workload by augmenting the basic teleoperation configuration with aids that emulate the natural visuo-motor system.



(a) The teleoperation configuration showing the information filters on the sensing and control capabilities of the operator.



(b) Teleoperation camera, gimbal, vehicle, and task element.

Figure 1.3: Teleoperation block diagram and physical representation.

The most common teleoperation configuration utilizes a live video feed to provide information about the remote environment as shown in Fig. 1.3. Two key considerations that need to be

addressed for the teleoperation configuration are situational awareness and pilot workload. To allow for adequate situational awareness, the camera must either have a large field of view or be controlled by the operator. During highly dynamic and interactive tasks, like operating a miniature rotorcraft or performing surgery, it is not feasible for the operator to manually adjust the camera while simultaneously performing the task. If a fixed camera with a wide field of view is used, problems can arise due to inadequate resolution in the area of interest or the need to operate outside the set field of view. The pilot workload while performing flight tasks can be high due fast acting dynamics and the unstable nature of the vehicle. Determining the appropriate level of automation can be critical for reducing workload without impacting performance.

In teleoperation applications the information flow is limited in both directions. Fig. 1.3(a) shows the components of a teleoperation system. The information filters G_{Head}, G_{Sense}, and $G_{Control}$ represent the fact that the operator's sensing and control capabilities are modified by the teleoperation infrastructure. On the remote side of the system, the control interceptor (like a joystick) is often the only input information available. With advances in eye tracking technology inexpensive systems are becoming available. Eye tracking measurements can be a rich source of information that can be exploited to improve the perceptual processes as well as providing knowledge on the dynamics of the remote vehicle. The gaze vector for the teleoperation system, shown in Fig. 1.3(b), is the combination of the gimbal orientation and the tracked eye position on the teleoperation display. Gaze provides a measurement of the human perceptual state and can be exploited to estimate the state of the controlled vehicle as well as the key environmental features of the remote system [12].

To improve teleoperation performance, models of the pilot's perceptual and control capabilities are identified and used to automate parts of the teleoperation task. First, the camera actuation in Fig. 1.3 is controlled using the operator's gaze patterns to achieve a natural visual display and improve situation awareness. The human eye contains a small area of high visual acuity that is relocated as needed to areas of interest through coordinated head-eye movements. The models for head-eye motion are identified using experiments with the operator visually tracking a moving laser point while wearing a pair of eye tracking glasses. The teleoperation configuration exploits these natural capabilities by generating movement of the remote camera that mimics head movements and simultaneously extracts gaze information relevant for control of the teleoperation task. A second improvement is to implement an augmented control system to assist the operator. The control augmentation generates the necessary signals to maneuver

the vehicle for a specific task with the operator only needing to provide corrective adjustments.

1.1.2 Human-in-the-loop Systems

For many operating scenarios fragile automation systems are unable to provide adequate performance. In contrast, human-in-the-loop systems demonstrate an ability to adapt to changing and complex environments, stability in control response, high level goal selection and planning, and the ability to perceive and process large amounts of information. During teleoperation tasks the human operator and the automated systems provide complementary capabilities with the human able to excel at high-level reasoning, task determination, spatial mapping, and guidance while the automated systems excel at dynamic regulation, trajectory optimization, and path following. Fitts [1] described the human-in-the-loop trade-offs and characterized the complementary capabilities of humans and machines. Table 1.1 lists the relevant strengths of each.

Humans are better at	Machines are better at
Stability in control response	Fast control response
Adaptation to changing environments	Repetitive and precise tasks
Pattern recognition and processing large amounts of data	Processing information
Inductive reasoning	Deductive reasoning
Performing when overloaded	Multi-tasking
High level goal selection and planning	

Table 1.1: Human-in-the-loop trade-offs [1].

Integration of a human operator and a remote vehicle requires overcoming significant technical challenges. Not only must the mechanical, electronic, communication, and interface systems all operate effectively, but in addition consideration needs to be given to the information received and transmitted by the operator. In order to better understand the information requirements it is necessary to characterize the control mechanisms employed, in particular the perceptual aspects for different control modalities including, regulation, tracking and more general guidance. Key questions include what type of information is required for the different control modalities and how the operator uses this information as part of the perceptual control mechanisms. The goal is to model the capabilities of the human pilot and then utilize these models to augment the control system. The augmented system reduces operator workload by moving the

operation from a continuous control mode where the pilot provides frequent adjustments to an intermittent mode that only requires occasional adjustments as shown in Fig. 1.4.

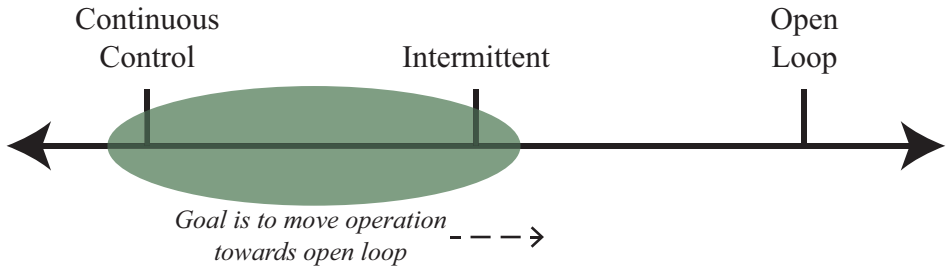


Figure 1.4: Problem characteristics.

The general approach in this work is to model the pilot's perception of information directly obtained from the environment. Studies based on simple synthetic cues [13] overlaid on the operator display, like numerical values for the vehicle state, may not capture the complexity of the pilot's perceptual processes. In real world applications, synthetic cues may not be available due to lack of sensing or might not represent the full set of information needed for control. In addition, synthetic cues typically show elements of the vehicle state and miss key interactions between the vehicle, environment, and task. Investigating perceptual information obtained directly from the environment allows real-world studies that are not limited to scenarios where adequate information can be overlaid on a display. The general approach also captures the embeddedness of the agent and more fully describes agent-environment interactions.

Human operation of vehicles while performing guidance and control tasks requires the operator to adapt to the environment while at the same time manipulating the system inputs to perform the desired task. Modeling the human's capabilities entails defining the information extracted from the environment through perception and also identifying the form of control response. To study the problem, the vehicle and environment need to be considered as two coupled dynamical systems. The adaptability and stability characteristics of the human operator emerge from the interaction of these two dynamical systems while taking into account the task and constraints.

1.2 Research Statement

The central research question investigated in this dissertation is: how does the human operator structure their control and guidance response? The approach is to gain an understanding of the deeper characteristics of human-vehicle-environment interactions through modeling and analysis techniques based on dynamic systems and controls. Secondary questions to be answered include:

- What is the representation of the information that is visually extracted by the human?
- How is the gaze used to estimate the state of the vehicle and task elements?
- How can the control action be modeled and identified?

The computational framework shown in Fig. 1.2 is proposed to capture the human perception-action interaction for teleoperated guidance and control tasks. The control theoretical view of human pilot modeling formalizes the pilot, vehicle and environment as a system that observes the current state, compares this state with a desired state, and then takes action to move the current state towards the desired one. This knowledge is also relevant for the development of interactive robotic systems. In particular, the concepts on perception and adaptability apply to any system where the dynamics of human interaction are critical. The work in this dissertation focuses on modeling the system components highlighted in Fig. 1.5. The non-highlighted blocks for the higher, navigation based level are being investigated in a separate research effort [14]. Once models are identified, the components are implemented as part of a teleoperation system in order to alleviate operator workload and provide a natural interaction that mimics human head motion.

The approach considered in this work focuses on tasks that require the operator to regulate closed-loop control for a process with a high-bandwidth dynamic response and track desired trajectories. Experiments were conducted on small scale rotorcraft in a static indoor environment. The primary source of information is visual with the operator gaze being the key measurement. It should also be noted that learning during multiple experimental runs is not investigated.

The basic approach is to model the low level perceptual process as part of the pilot's feedback control mechanism. Historical attempts at modeling low level pilot control typically addressed single-axes problems. They also used simulated environments as opposed to real vehicles and only used low-level perceptual cues as feedback. The objectives of this work differ

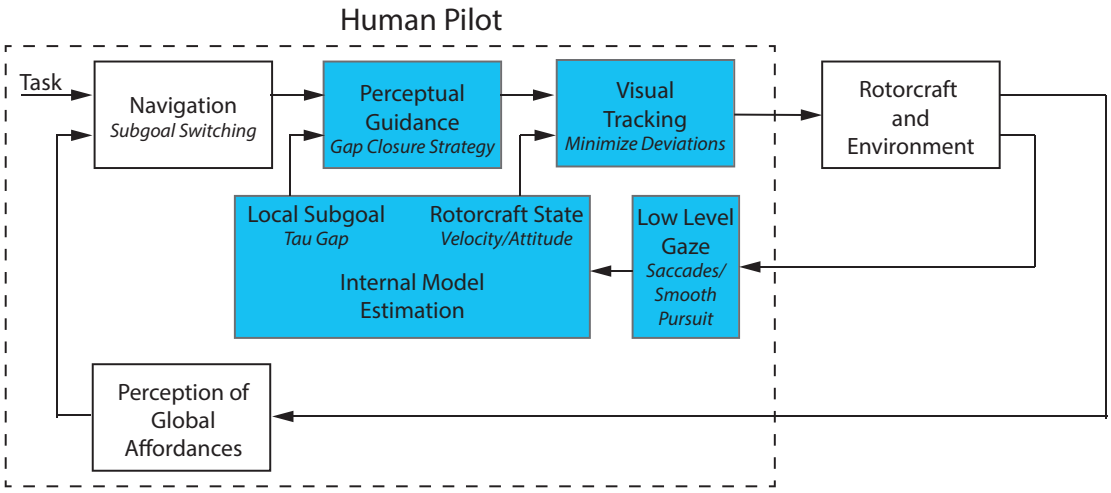


Figure 1.5: Research focused on highlighted blocks.

from existing research in several key ways. First, this work takes what have largely been conceptual ideas on action-perception and structures them to be implemented for a real world problem. Second, this dissertation investigates the perceptual side of the operator modeling problem and looks at how the perception is coupled with low-level dynamic control. Third, the approach utilizes the operator gaze to integrate visual action-relevant information with dynamic control models of responses. Finally, the proposed approach uses a body centric spherical reference frame that corresponds to human visual perception.

The goals of the research are summarized as follows:

- Build the infrastructure to capture the human perception-action interaction for teleoperated guidance and control tasks
- Model the relevant human-in-the-loop perception and action components
  - Decompose gaze patterns to provide information that can be incorporated into a control strategy
  - Estimate the vehicle state and task elements
  - Use a multi-loop closed loop control architecture to model the human actions
- Apply the models for human perception and control to a relevant real-world problem

1.3 Contributions

The contributions of this dissertation:

- Integrated the hardware and software infrastructure that captures measures of the human and vehicle state and allows for the study of human/vehicle interaction.
- Established the methodology to collect and register the data needed to identify components of the human perception/control models that includes operator control inputs, eye tracking measurements, head orientation, and the motion of the test vehicle. Conducted experiments with human test pilots and collected data for basic dynamic control tasks.
- Proposed the model structure that integrates human perception, internal state estimation, trajectory generation, and control.
  - Investigated a novel approach for utilizing gaze as the primary sensory mechanism for measuring vehicle state and task elements.
  - Determined a representation of the human pilots internal model of the vehicle state and task elements (Internal Model Representation block in Fig. 1.5).
  - Characterized the pilot's control response by identifying models for the Perceptual Guidance and Visual Tracking blocks shown in Fig. 1.5.
- Implemented natural augmentations for teleoperation to simplify perception and control for the human pilot and demonstrated the benefits.
  - Demonstrated a novel application that uses gaze to automate positioning control of a remote camera based on models of the human head/eye system.
  - Automated control actions while still allowing pilot input to reduce pilot workload and improve task performance.

1.4 Dissertation Organization

The dissertation outline is given in Fig. 1.6. As shown in the figure, the dissertation is organized into background information, description of the experimental environment, modeling of the human perception-action capabilities, and an application example. The description of the contributions of each chapter is given below:

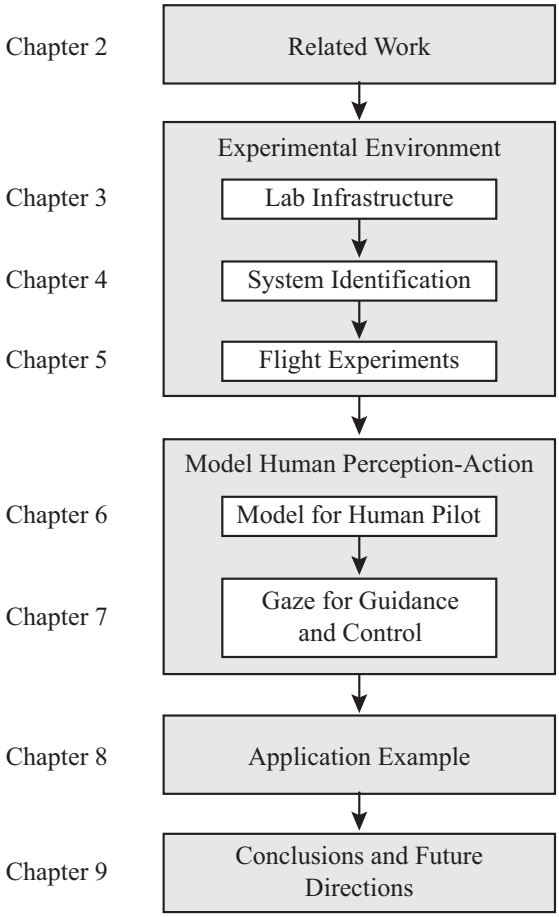


Figure 1.6: Dissertation outline.

- **Chapter 2**

  Related work is reviewed that includes general cognitive models, human perception-action, control theoretic models of human performance, human guidance behavior, eye-head coordination and gaze, and multi-loop control analysis.

- **Chapter 3**

  To investigate the human-in-the-loop perception and control processes a lab environment capable of capturing the relevant information is required. This chapter describes the hardware and software systems that capture the operator's eye focus, head motion, and control actions. This data is synchronized with motion information on the experimental vehicle. The systems for implementing remote teleoperation are also detailed.

- **Chapter 4**

  The primary approach for understanding the human-vehicle interactions uses formal modeling and analysis techniques from dynamic system identification and controls. For the lab environment the dynamic hardware elements that need to be modeled are the helicopter and the actuated camera gimbal. This chapter provides the details on the system identification method and the resulting dynamic models.

- **Chapter 5**

  The objective of the flight experiments described in this chapter is to provide data for identifying the structure of the human operator's control response and patterns of gaze that allow for the completion of a given task. Three experimental tasks are investigated that involve stationary flight, linear motion, and circular motion. The tasks each emphasize different aspects of the human perception-action modeling problem. Metrics to quantify the pilot's performance are also defined.

- **Chapter 6**

  In this chapter human control response is modeled as a trajectory generation component along with feed-forward and feedback elements. This chapter discusses the Perceptual Guidance and Visual Tracking blocks shown in Fig. 1.5. The structure of the low level control loop and the control parameters are identified for linear and curvilinear tasks. An approach based on Tau theory is used to generate the desired trajectories for the Perceptual Guidance block.

- **Chapter 7**

  The objective of this chapter is to model the role and contribution of the operator's gaze motion in remote control operation of an agile vehicle (the Low Level Gaze and Internal Model Estimation blocks in Fig. 1.5). The overall human-machine system is described using a multi-loop manual control model. The sensing requirements for each loop are established by investigating the relationship between the operator's visual gaze trajectories, the vehicle trajectories, and the control actions. Lastly, a state estimation approach incorporates gaze information to estimate the vehicle and environmental features.

- **Chapter 8**

  An example application is presented that automates the positioning of a remote camera

based on the operator gaze behavior and augments the flight control system to simplify the task for the pilot. The camera is mounted on an actuated gimbal that uses real-time gaze measurements to mimic human head movement. The control architecture is built on models of head-eye coordination to provide natural head-eye-scene interactions. Control augmentations to reduce operator workload are also described with experimental results demonstrating the effectiveness of the approach.

- **Chapter 9**
  In the chapter key conclusions are presented along with possible avenues for future research.

- **Appendix A**
  Gives an example of a traditional approach to vision based estimation.

1.5 List of Publications by Chapter

Parts of the content for the chapters of this dissertation have first appeared as various publications. Below are the publications corresponding to each of the chapters:

- **Chapter 1 and 2**
  B. Mettler, Z. Kong, B. Li and J. Andersh, "Systems View on Spatial Planning and Perception Based on Invariants in Agent-Environment Dynamics", *Frontiers in Neuroscience*, 2014.
  B. Mettler, J. Andersh, and N. Papanikolopoulos, "A First Investigation into the Teleoperation of a Miniature Rotorcraft", Springer Tracts in Advanced Robotics, *Experimental Robotics*, vol. 54, pp. 191-199, 2009.

- **Chapter 3**
  J. Andersh, B. Mettler, and N. Papanikolopoulos, "Miniature embedded rotorcraft platform for aerial teleoperation experiments", *17th Mediterranean Conference on Control and Automation*, 2009.
  J. Andersh and B. Mettler. "System integration of a miniature rotorcraft for aerial teleoperation research". *Mechatronics*, vol. 21, no. 5, pp. 776788, 2011.
  B. Mettler, N. Dadkhah, Z. Kong, and J. Andersh, "Infrastructure and Research Framework for Human Interactive and Autonomous Guidance", *International Conference on*

Unmanned Aircraft Systems (ICUAS'12), Philadelphia, PA, 2012.

B. Mettler, N. Dadkhah, Z. Kong, and J. Andersh, "Research Infrastructure for Interactive Human- and Autonomous Guidance", *Journal of Intelligent & Robotic Systems*, September, 2012.

- **Chapter 4**

  J. Andersh and B. Mettler, "Report: System Identification for a Blade mCX Micro Helicopter", University of Minnesota, 2014.

- **Chapter 5**

  J. Andersh, B. Mettler, and N. Papanikolopoulos, "Experimental investigation of teleoperation performance for miniature rotorcraft", *In 48th IEEE Conference on Decision and Control*, December 2009.

- **Chapter 6**

  B. Mettler and J. Andersh, "Fundamental Control Characteristics of Curvilinear Motion in Human and Automatic Path Tracking Tasks", *2013 American Control Conference*, pp. 6460-6467, 2013.

- **Chapter 7 and 8**

  J. Andersh, B. Li, and B. Mettler, "Modeling visuo-motor control and guidance functions in remote-control operation", *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014.

  B. Li, B. Mettler, and J. Andersh, "Classification of Human Gaze in Spatial Guidance and Control", In *Systems, Man, and Cybernetics (SMC), 2015 IEEE International Conference on*, pp. 1073-1080, 2015.

- **Appendix**

  J. Andersh, A. Cherian, B. Mettler, and N. Papanikolopoulos, "A vision based ensemble approach to velocity estimation for miniature rotorcraft", *Autonomous Robots*, vol. 39, no. 2, pp. 123-138, 2015.

Chapter 2

Related Work

2.1 Introduction

This dissertation models the perceptual and control processes of a human operator. Fig. 2.1 shows the different research areas that provide background for the work. The first section of this chapter considers general approaches for modeling a human's behavior and perception. Following sections focus on specific aspects of the larger problem such as guidance behavior, perceptual guidance, human control models, and human gaze.

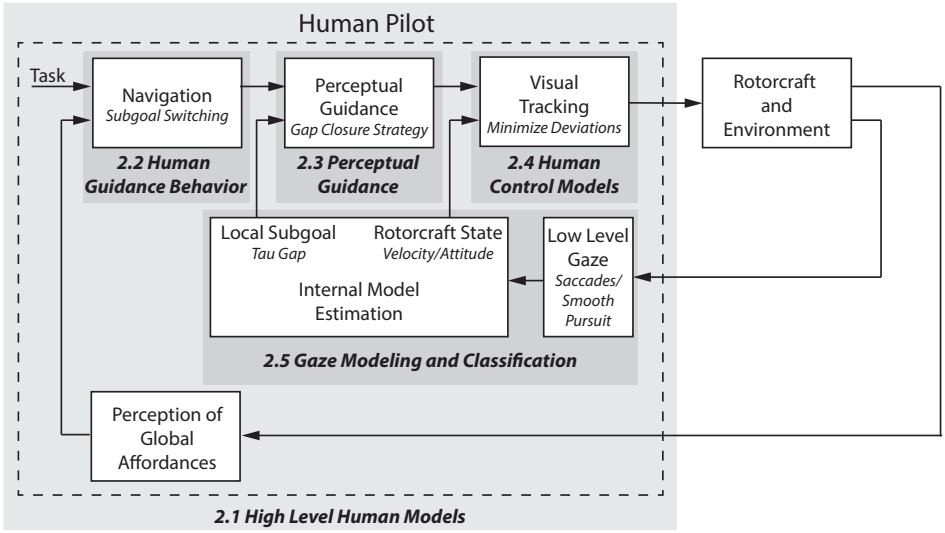


Figure 2.1: Related work sections and how they correspond to the proposed model structure.

2.2 High-Level Human Models

In the literature, cognitive models have been developed by researchers from a variety of disciplines including psychology, computer science, robotics, human-computer interaction, cognitive science, neuroscience, and human factors engineering. These models are high level conceptual constructions intended to cover a broad range of human behavior. These models propose high level structures for human processing but lack many of the necessary details to implement for real world scenarios. This section describes the key work done in relevant research areas.

2.2.1 General Cognitive Models

Cognitive models have been proposed to cover a range of domains and applications. Examples include the Model Human Processor by Card [15] and Perceptual Control Theory described by Powers [16]. Card drew an analogy between a computer's capabilities and a human's memory, motor, and perceptual abilities. The approach modeled the cognitive and motor processing times for a task through experimentation. The purpose of the model was to predict the time it would take a user to complete a task during the design process. This work follows an idea similar to Card in that individual model components are thought of as processing elements that can be modeled. The Perceptual Control Theory model described by Powers is built around the idea that human's use feedback to adjust their actions so that the desired perceptions are achieved. The catch-phrase of the approach is "behavior is the control of perception." The idea that human actions can be modeled as a nested series of feedback loops can be seen in Fig. 2.1.

Situation awareness (SA) as outlined by Endsley [11] involves knowledge of what is going on in the environment and what is important with respect to the current task. SA concepts are inherent in the proposed hierarchical control architecture in Fig. 2.1. The amount of SA needed to complete a task can be defined as the aggregate of all the information found by the perceptual processes (desired states and actual states) for the different levels in the hierarchy. Endsley described the components that make up the definition of SA. The first component, perception of elements in the environment, is clearly part of the perceptual processes in the hierarchy. The second component, comprehension, can be seen in the levels that perform subgoal selection and trajectory generation. The final component, projection or prediction, is seen in all the levels.

Action regulation for complex systems was discussed by Dörner [17]. The work breaks

down the process into phases including goal elaboration, hypothesis formation, prognoses, planning, monitoring, and self reflection. Errors corresponding to each phase are described along with potential reasons for the mistakes. Albus [18] proposed a multi-scale planning model that used a hierarchical structure to model human response. In this approach the abstraction of the representation increases with higher levels while the resolution decreases. Both Dörner and Albus provide concepts that inherent to the model structure in Fig. 2.1. A final relevant model was proposed in [19]. In the model the levels in a nested hierarchy have increasing bandwidth when moving from outer to inner loops.

Pew [20] discussed the structure of human perceptual-motor performance and identified three levels of organization. The lowest level acts as a simple servomechanism that generates motor outputs to correct differences between the perceived and desired state. The next level captures the human capacity to identify and implement patterns of motion based on the predictability of task and environment. The final level considers the goal and environment to call from memory integrated patterns of movement. The structure defined by Pew is similar to the approach utilized in this work. However, this work takes the conceptual ideas and seeks to identify detailed perceptual and control models.

2.2.2 Human Perception-Action

Gibson's school of ecological psychology was the first to emphasize the agent-environment coupling [21]. The ecological approach to perception described a dependence between the operator's control response and perception. The control response is driven by the perceived state of the vehicle and environment, while perception is largely defined by the movement resulting from control actions. Consequently, attempting to study the perception and action problem by focusing on either perception or action alone only captures part of the problem. To truly model the human operator and capture the inherent flexibility and stability requires an approach that investigates both.

Gibson also coined the term affordance to represent a features of the environment that present an opportunity for action [22]. Investigation into perceptual aspects of affordances include work on the accuracy of affordance perception [23], relation to body dimensions [24], and affordances that account for movement capability [25]. Using affordances as part of a control strategy to guide action was discussed in [26]

Based on the ecological psychology movement, research interest in a more formal dynamics and control-based theory of perception and action has grown. Warren proposed a simple model of behavior dynamics that describes the agent and environment using dynamical systems theory [27, 28]. Warren's approach integrates four main ideas: (i) the agent is embedded in the environment, (ii) control is based on information about the agent-environment state, (iii) control actions are specific to the current task, and (iv) behaviors result from agent-environment interactions. Fig. 2.2 shows the basic dynamical systems describing the agent and environment with **e** representing the environment state, **x** the vehicle state, **a** the human state, and u the control action. The functions g, f, and k give the dynamics of the environment, vehicle, and human respectively. Information is extracted from the environment with the vector i capturing the information variables and λ defining the function that converts the environment state to human's internal representation. Finally, the control action u results from the control law β. This embedded closed-loop perception-action cycle leaves many details to be elucidated regarding the information extraction laws and type of control structures and their associated input, output and state variables.

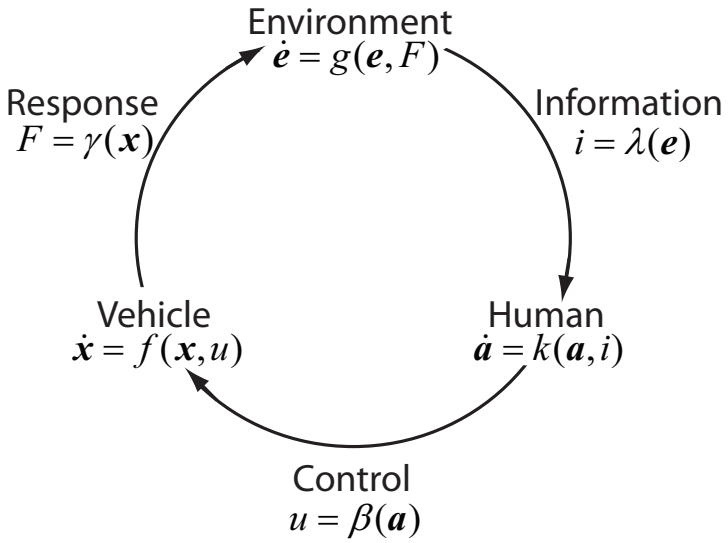


Figure 2.2: Dynamics of the perception-action cycle.

Applications of this model have emphasized the role of dynamics in coordination, however, results are mostly limited to simple tasks such as balancing an object [29], bouncing a ball on a racket [30, 31], intercepting a moving target [32], or walking [33, 34, 35]. A well-known

example is the catching of a fly ball by a baseball outfielder [36].

2.3 Human Guidance Behavior

Guidance includes a range of dynamical interactions, starting with the vehicle or body itself, and then extending into the dynamics that encompass the entire human-machine or agent-environment [37]. When humans operate in natural environments, such as a piloting in complex terrains or performing surgery, they have to learn the patterns of interaction between the environment and motion, as well as learn to extract useful visual cues. In [38] a mapping technique was introduced to study the spatial characteristics of ensembles of trajectories collected from precision interception experiments. *Interaction Patterns* (IPs), which are structural features emerging from the dynamical interactions in the agent-environment system, have been recently proposed as a way to formalize these concepts [38]. The IPs let a human organize their behavior in ways that mitigate the various sources of complexity. Invariants in this larger system are expected to play a central role in shaping the architecture responsible for integrating controls, perception and planning processes. These results were integrated under a hierarchic model in [39].

2.4 Perceptual Guidance

Models to explain perceptual guidance have been investigated for both animal and human guidance behavior. The most widely accepted approach, called tau theory, originates from Gibson's ecological psychology and was proposed by Lee [40]. The central idea is that the visual and motor systems form a unified perceptuo-motor system. The system utilizes a biological variable τ that represents the time to contact at the current closure rate. In the simplest form $\tau = y/\dot{y}$, where y is the motion gap and $\dot{y}$ the rate of closure of the gap. An example of how τ is calculated is shown in Fig. 2.3. One of the main benefits of the theory is that the τs are naturally extracted by the visual system. Another strength of this mechanism is its simplicity, which enables real-time implementation. The τs are generally not measured directly but manifest as invariants in the flow field. A general shortcoming of perceptual theories, however, is that they do not provide an understanding of the larger planning and reasoning processes. The theory was extended to include the concept of an intrinsic action gap generated internally and how that

is coupled of the physical action gap [41].

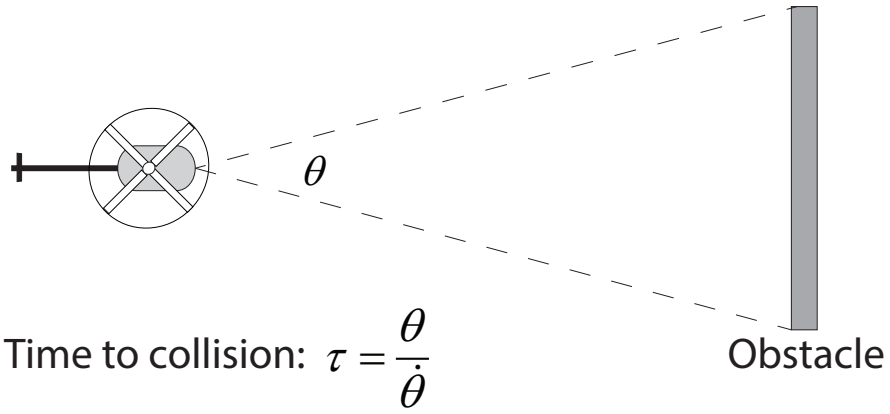


Figure 2.3: Example of τ for a helicopter moving towards an obstacle.

Tau theory has been verified for numerous simple control tasks in humans and animals. Examples include bird landing [42], hummingbirds docking on a feeder [43], foot landing during long jumping [44], and drivers braking [40]. More challenging examples involving the coordination between motion in multiple dimensions as well as the incorporation of tau guides are discussed in [41]. Recent work has utilized the theory for investigating helicopter pilot behavior [45, 46, 47].

Visual attention in the context of guidance behavior represents a recent research direction. Significant progress has been made understanding insect sensory guidance systems [48, 49], however these creatures occupy significantly different environments than humans, and are confronted with more restricted planning and adaptation challenges. In humans, attention models have previously focused on still images (see for e.g. the benchmark eye movement datasets [50]). Itti describes a computational model of attention that incorporates saliency, inhibition of return, coupling with eye movements, and scene understanding [51].

2.5 Human Control Models

Since the middle of the last century researchers have worked to model human control actions, especially for the case of piloting an aircraft. This section details work on characterizing the human pilot and discusses how the response can be modeled as a multi-loop system.

2.5.1 Control Theoretic Models of Human Performance

Pilot modeling has traditionally been treated from three different perspectives: control theoretic modeling, psychophysical modeling and biomechanical modeling [52]. Research efforts towards modeling the human as a controller first began in the 1940-1950s to study human motor performance with the first significant publication coming from Tustin in 1947 [53]. The research investigated the manual control response of an operator targeting a gun turret. The main contribution of the work was to demonstrate that a linear control law with remnant could describe the operator response. Russell extended the work on human modeling by devising an experimental method that simulated different system dynamics while measuring the operator control action [54]. The work demonstrated that a human can adapt their response to compensate for varying system dynamics. Elkind performed experiments using a wide variety of inputs constructed from a number of sine waves of different amplitude and frequency [55]. From the data, transfer functions were identified that covered a wide range of system characteristics.

One of the primary results in the field of operator performance modeling was the work of McRuer on the crossover-model [56]. It was shown that for systems like the one seen in Fig. 2.4, the combination of the human operator (Y_p) and the system dynamics (Y_h) can be approximated by a simple integrator with delay system near the crossover frequency ω_c [57].

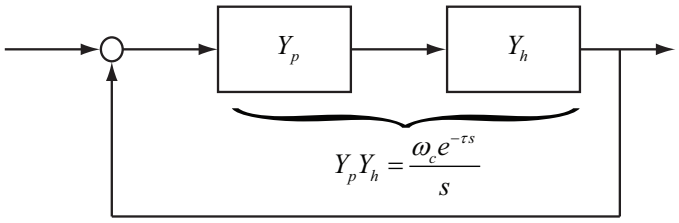


Figure 2.4: Block diagram of the general tracking task in one dimension.

McRuer continued work in the area developing the quasi-linear control model [56]. Based on the cross-over model and linear control models McRuer identified the transfer function of the human operator for a number of system dynamics types [58, 59]. The work showed that the same general loop transfer function $L(s) = \omega_c/s$ was valid for a number of situations with the human operator adapting performance to compensate for the system dynamics. These models focus on tracking and pursuit tasks in which subjects track a given visual stimuli.

Kleinman and Baron developed a more advanced human performance model based on optimal control theory [60]. The theory assumes that a motivated and well trained operator performs like an optimal controller. The main benefits of Kleinman's optimal control model include the ability to deal with multi-variable systems and a close association with human information processing principles.

A more recent control model of the human pilot was the structural model proposed by Hess [61]. The goal of the structural model was to incorporates a more realistic representation of human signal processing. The model shown in Fig. 2.5 captures the open loop neuromuscular system, the operator's internal model, the display, and the vehicle dynamics. The structural model has been utilized to study a number of pilot characteristics including pulsive control [62], preview control [63], pursuit control [64], and vehicular control [3, 65].

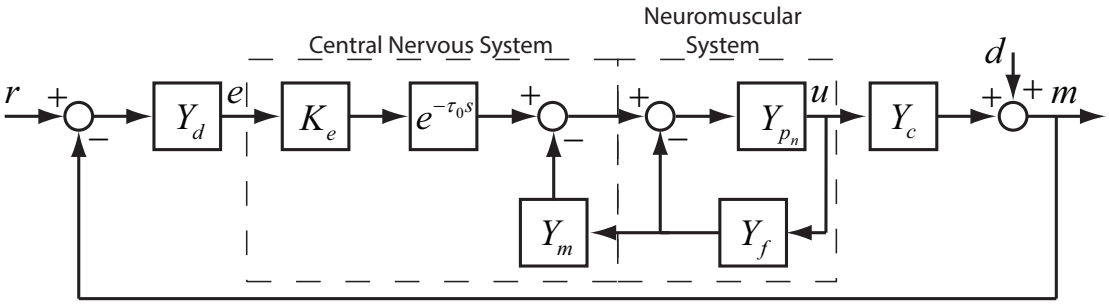


Figure 2.5: Hess's structural model of the human pilot [3].

2.5.2 Multi-loop Control Analysis

In the 1960s Krendel introduced the Successive Organization of Perception (SOP) framework [66]. The framework described a progression of human control skill that starts out as compensatory, moves to a pursuit type organization, and finishes as an open-loop response. SOP describes the human internal processes that develop as skills improve. The idea also can be seen as identifying the structural blocks necessary to capture the control response of a human pilot, namely the compensatory, tracking, and open-loop components.

The multi-loop pilot model is essentially a form of embedded agent-environment model. It describes the human control behavior in terms of a nested series of control loops of increasing bandwidth and was proposed to describe pilots' manual control [67, 68, 69]. The loops are organized hierarchically as shown in Fig. 2.6 starting with the low-level attitude stabilization.

Next, a guidance element generates trajectories to achieve the desired objective. Finally, the navigation block performs goal identification and directs high level maneuvering.

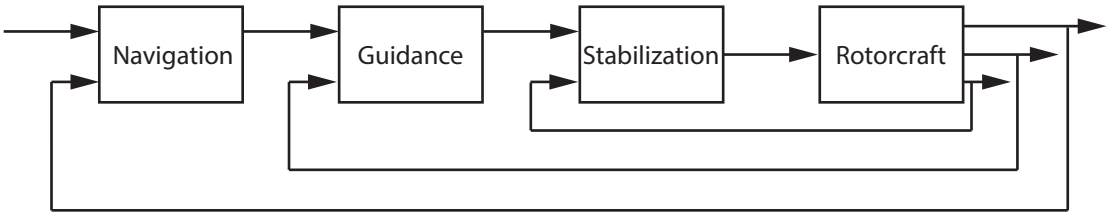


Figure 2.6: Multi-loop pilot model structure.

Analysis of pilot performance based on a integrated pilot-vehicle multi-loop model for helicopter maneuvers was proposed by Heffley [70]. The pilot modeling work was extended into the Adaptive Pilot Model (APM) by Padfield [4]. The APM is based on the concept that the pilot converts the complex coupled pilot-vehicle system to a simple relationship between command and control output. The model utilizes a multi-loop architecture to capture the pilot response. Fig. 2.7 shows the block diagram of the APM model. For a translation command X_c the pilot generates an attitude θ that moves the vehicle to X based on vehicle dynamics $Y_{A\theta}$. The pilot model to convert the position error X_e to an attitude command θ_c is given by Y_{PT}. The pilot model for the attitude error is given by $Y_{P\theta}$.

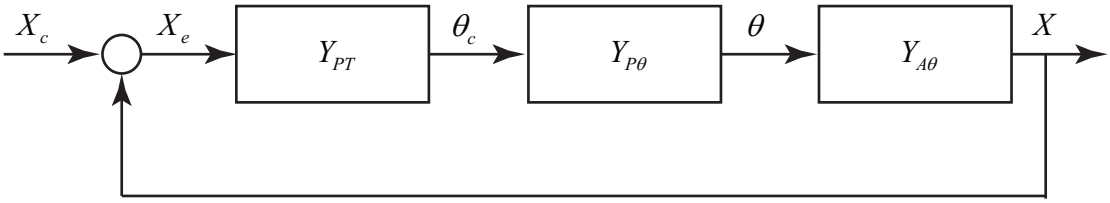


Figure 2.7: Adaptive pilot model block diagram [4].

2.6 Gaze Modeling and Classification

Human vision provides the primary source of information for humans everyday activities, from ordinary behaviors like reading, walking and driving [71, 72], to highly specialized tasks like surgery, tele-operation and sports. Visual perception is achieved via the deployment of a foveated

visual system [73]. The fovea spans a small optical angle in the visual field where high resolution information is acquired. Humans extract knowledge about the environment by actively orienting the fovea with the coordination of eye movement and head movement. this coordinated eye-head motion is called gaze. This section looks at work on modeling the eye-head motion as well as identifying the different gaze modes of operation.

2.6.1 Gaze Models Based on Eye-Head Coordination

Gaze movement is the transition of visual focus in space, which involves both eye movement relative to head, and head movement relative to space [74]. That is, gaze control encompasses the entire eye-head coordination, which is attenuated by the vestibulo-ocular reflex (VOR). The eye-head coordination has been frequently investigated. Bizzi [75] proposed that the eye movement is programmed based on the planned head movement. When head movement is simultaneously adopted, it reduces the magnitude of eye movement. In contrast, Guitton and Volle [74] suggested that the gaze control system can utilize all available components synergistically. More specifically, Wijayasinghe, et al [76] pointed out that the VOR would rotate the eye backward to compensate for the forward movement of the head, minimizing the cost.

Head-eye coordination during gaze shifts has been mainly investigated in the psychophysics field. Models explaining gaze orienting to targets have been determined for both 'within' and 'outside' the occulomotor range [77, 74, 5]. Fig. 2.8 shows the model based on [5]. The model describes the combined head-eye dynamics during *gaze shifts*. Two aspect of the model that are noteworthy are the independent control of the head and motor systems, i.e., head motion can be controlled as a separate system, and, that initiation of eye and head movement is controlled by different gating mechanisms.

In addition to *gaze shifts*, the *smooth pursuit* eye tracking mode is also relevant for teleoperation systems. Smooth pursuit occurs when the operator is visually tracking a moving stimulus. During smooth pursuit the eye remains focused on the moving object with the head-eye motor system coordinating the gaze motion. Fig. 2.9 describes a closed-loop structure proposed by Lisberger that generates the smooth pursuit response [6].

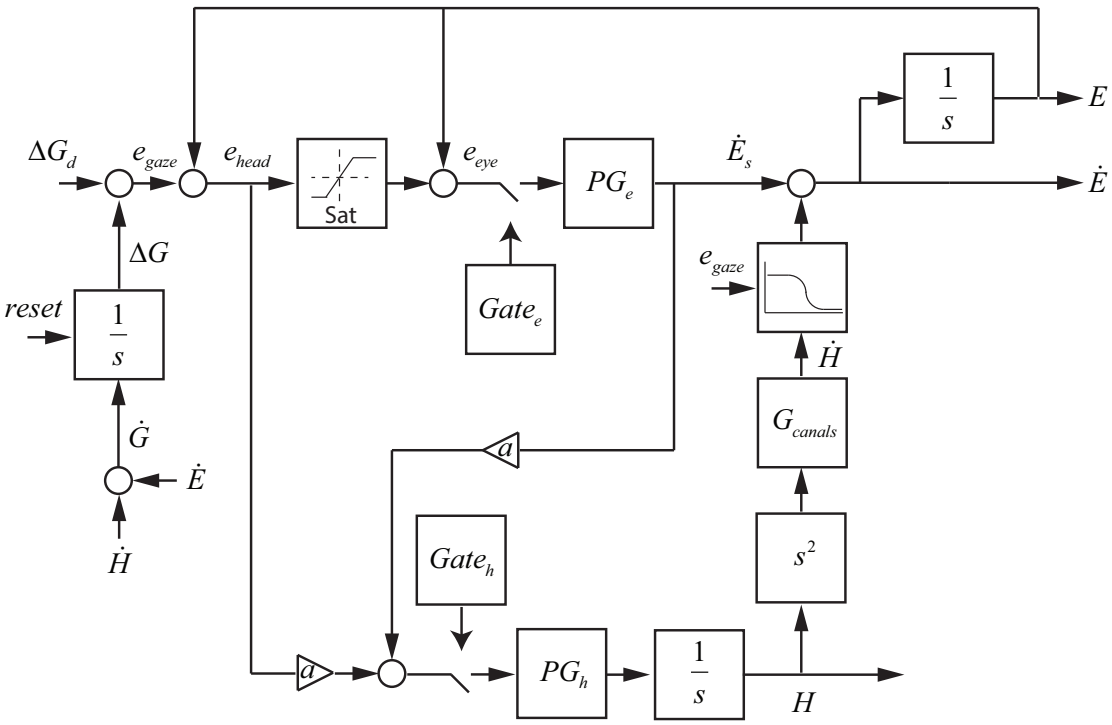


Figure 2.8: Model of head-eye coordination during gaze shifts [5]. The variables e_{gaze}, e_{head}, and e_{eye} are the gaze, head, and eye errors. The gaze variables ΔG_d, ΔG, and $\dot{G}$ are the desired, current, and velocity respectively. The eye position and velocity are given by E and $\dot{E}$ with the saccadic eye velocity being $\dot{E}_s$. Head position and velocity are H and $\dot{H}$.

2.6.2 Gaze Classification

The vital role of vision explains the rising interest in recording gaze movement. Young and Sheena [78] reviewed techniques for measuring eye movements, and categorized principal approaches with respect to the associated eye characteristics, including electro-oculography, corneal reflection, limbus-pupil-and-eyelid tracking, and contact lens methods, etc. Recent developments extend the range of application for eye measurement systems with high sampling frequency, high precision, low cost and no contact.

Recent advancements in eye tracking systems technology enables the study of the mechanism of active gaze movements during diverse tasks and conditions. Land [71] examined the ways that vision contributes to behavior organization, and concluded that humans use gaze proactively seeking out the information required in the seconds before the execution of actions.

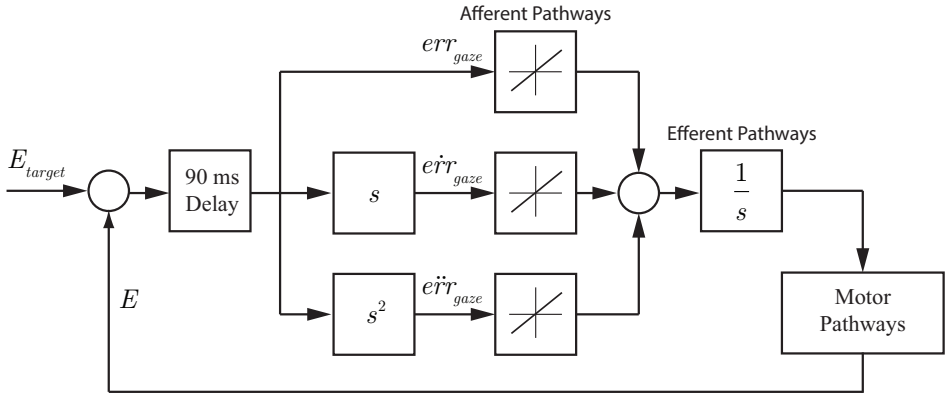


Figure 2.9: Model of head-eye coordination during smooth pursuit [6]. The variables E and E_{target} are the current and desired eye positions. The retinal position, velocity, and acceleration errors are given by err_{gaze}, $\dot{err}_{gaze}$, and $\ddot{err}_{gaze}$.

Hayhoe and Ballard [72] demonstrated that gaze movements are driven by prospects of reward. In a study of guidance tasks, Mettler [79] found evidence that human pilots direct gaze to identify the most information-rich visual cues (in terms of subgoals) as part of the trajectory planning process.

Gaze classification has been realized based on the distinction in the kinematics of the three gaze patterns, i.e., by setting respective thresholds of velocity and duration range. Salvucci and Goldberg [80] focused on fixation identification and proposed a taxonomy of classification algorithms with respect to spatial and temporal characteristics, including Velocity-Threshold Identification (I-VT), Hidden Markov Model Identification (I-HMM), Dispersion-Threshold Identification (I-DT), Minimum Spanning Tree Identification (I-MST), and Area-of-Interest Identification (I-AOI). Komogortsev and colleagues [81] extended this research by introducing another algorithm, Kalman Filter Identification (I-KF), and provided recommendations for the selection of the input parameters for each algorithm.

The three gaze patterns (fixations, saccades, and smooth pursuit) have been widely accepted to provide essential insights into gaze movement. It is only during fixations and smooth pursuits that high quality visual information is acquired. Smooth pursuits are used to update the dynamical state information needed for regulation [12]. Fixations are tightly linked in time to the evolution of the task [72]. High velocity and short duration of saccades render the visual system essentially blind, but they also reflect the economy of human attention organization [82].

The performance of classification can be interfered with by several factors. First, the above methods only classify gaze based on *eye movement*, while gaze movement needs to integrate head pose information that is always recorded in a different coordinate frame from that of eye movements. Specifically, head poses are in the inertial Cartesian frame and eye movements are on the image plane. Therefore, conversion is required in order to register gaze positions in space. Moreover, micro-saccades can also degrade gaze classification. These are low-amplitude saccades employed during fixations and smooth pursuits as a correction mechanism that stabilizes the retina and prevents motion blur [83]. A successful gaze classification algorithm needs to consider both of these factors.

Chapter 3

Experimental Infrastructure

3.1 Introduction

This chapter describes an integrated research environment specifically developed to exercise and investigate guidance and control capabilities under human control, autonomous control, and augmented control modalities. The lab facility is designed to implement tasks that emphasize agent-environment interactions. The overall goal is to characterize these interactions and to apply the gained knowledge to determine models of the underlying perceptual and control processes. These models can then be used to design guidance and control algorithms as well as human-machine systems. The facility uses miniature rotorcraft as test vehicles with a Vicon motion tracking system and SensoMotoric gaze tracking system. Fig. 3.1 shows the proposed model structure with the lab infrastructure described in this chapter highlighted.

3.2 Experimental Overview

The primary objective for the facility is to support experimental investigation of the deeper characteristics of agent-environment interactions. The approach is to combine data driven methods with theoretical investigation through the application of formal modeling and analysis techniques from dynamics and controls. The knowledge can then be used to improve the design of algorithms for autonomous systems as well as interfaces for human-machine systems.

One difference between this work and other investigations of miniature rotorcraft operation is the focus on teleoperation and human factors. Most current research efforts are primarily

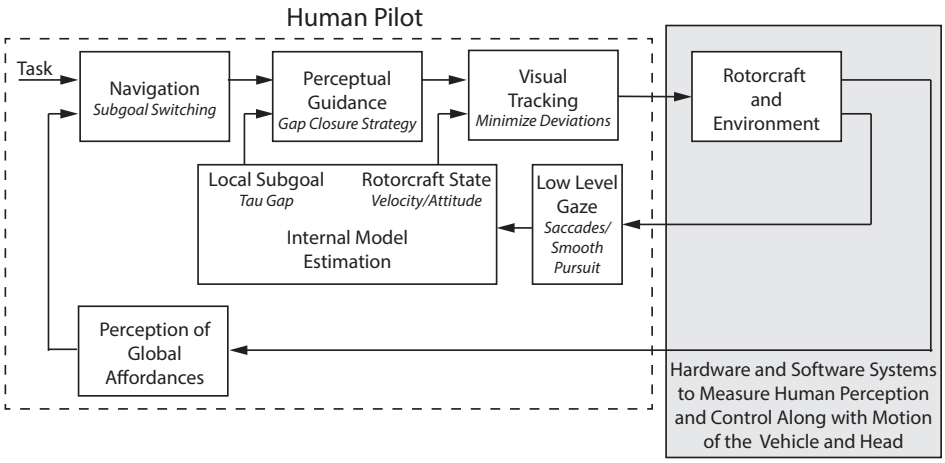
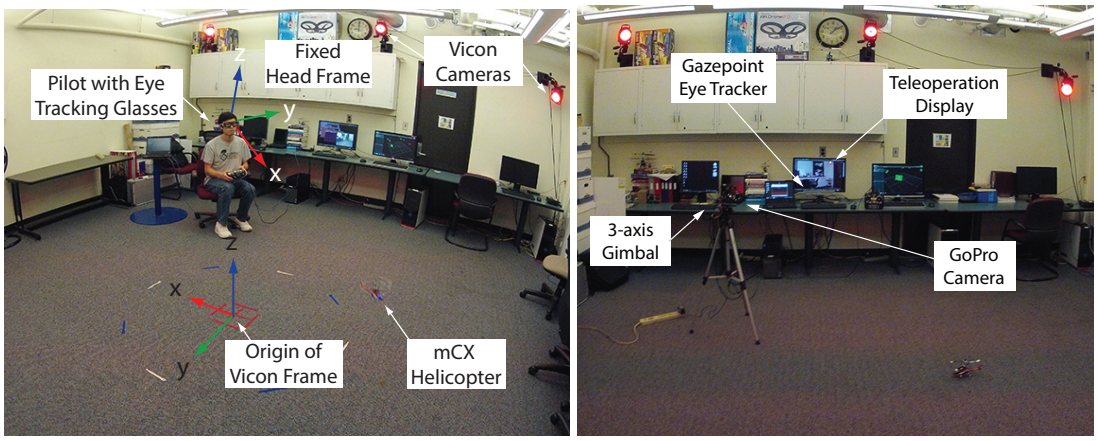


Figure 3.1: Model structure with the lab infrastructure highlighted.

focused on the development of the estimation and control systems needed for autonomous flight. Estimation and control are critical components of the overall platform described in this chapter, but they are considered from the standpoint of how they interact with a human pilot. The research enabled by the environment seeks to determine how spatial information is perceived while performing various tasks such as navigation and precision maneuvering. Modeling this type of perception and the corresponding control response of the pilot can give insight into what control modes are best suited for a given task. By understanding the limitations and capabilities of the human, the best suited control mode can be applied along with a level of control authority suitable for the task.

The research agenda requires being able to run experiments with actual hardware components that combine the effects of vehicle dynamics, environmental sensing, and measuring the human's perception and action. The lab facility was setup to use small-scale rotorcraft UAVs due to their maneuverability and compact sizes. The rotorcraft play as much the role of tool as the role of a flight vehicle. Fig. 3.2(a) shows an overview of the lab infrastructure where the pilot operates in a "third-person" perspective, i.e., the operator views both the vehicle and task from outside. A camera mounted on an actuated gimbal provides a teleoperation setup with a "third-person" perspective as seen in Fig. 3.2(b). A final experimental configuration uses an onboard camera to generate a live video stream that can be used to create a "first-person" ground station perspective.

The goal is to be able to run similar experiments under human or autonomous control and



(a) Third Person Perspective Direct View (b) Third Person Perspective Teleoperation View

Figure 3.2: Overview pictures showing the lab setup for flight experiments with pilot (a) directly observing the rotorcraft (b) operating in a teleoperation configuration.

record the relevant data ranging from vehicle motion to the control inputs and on-board sensors (camera, IMU, etc.). In parallel, the key components of the lab facility are modeled so that they can be integrated in a high-fidelity simulation with detailed photo-realistic graphical models of the lab facility and the Akerman Hall [84].

For experiments with human subjects, measurement of the operator control inputs, head pose, visual gaze, and field of view video are collected along the vehicle motion to provide data for the investigation of the control and perceptual functions. The experimental procedure and methods rely on collection of ensembles of trajectories. Using ensembles allows sampling of the human behavior over larger domains and thus provides a means to extract information about the larger strategies used for planning. Fig. 3.3 illustrates the main components used for human guidance experiments.

The software environment is designed to support the combination of heterogeneous processes and data from a variety of sensors. The system is integrated using the Robotics Operating System (ROS) [85]. ROS provides the flexibility to integrate processes through message passing. Also, ROS's open software environment helps to disseminate software to and from the larger robotics community and stimulates the creation of common standards.

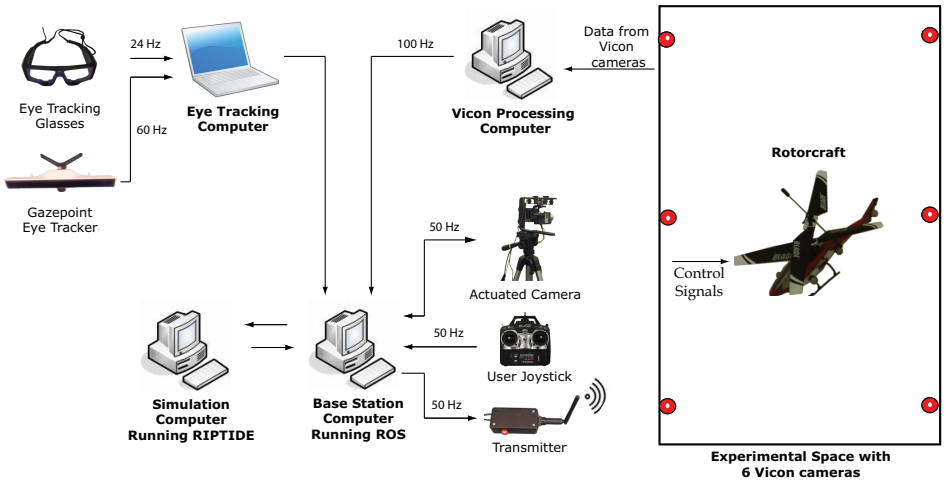


Figure 3.3: Architecture of the lab facility including computing, electronic and flight hardware.

3.3 Lab Hardware Components

The blockdiagram in Fig. 3.3 gives an overview of the key hardware components that make up the lab facility. The hardware components include joystick user inputs, direct transmission of control signals to Spektrum RC receivers, real-time readings from a Vicon motion tracking system, and integration with onboard systems. Additional functionality includes an eye tracking device for human perceptual experiments, a Kinect sensor for depth data and a simulation environment built using the RIPTIDE software developed at NASA. Table 3.1 describes the type of measurements that can be obtained from the system. This section provides a description of these hardware components and their usage.

3.3.1 Vicon Motion Tracking Cameras

The measurement and recording of the position and orientation of the aerial vehicles in flight is performed by a Vicon motion tracking system [86]. The motion tracking system consists of 6 high-speed MX-40 cameras that are affixed to the walls of the lab. The cameras are connected to a router and from there to a computer via an Ethernet link. Retro-reflective spherical markers are placed on the vehicle to form a shape that can be uniquely identified. Once the Vicon tracking system software (Vicon Tracker) has identified the helicopter as a tracked object, the position and orientation information can be streamed over a network connection as shown in

Data Source	Type of Information	Data Rate
Vicon	Pose of each object being tracked	100 Hz
Joystick	4-axis control inputs from operator	100 Hz
Onboard IMU	3 angular rates and 3 accelerations	100 Hz
Onboard camera	Images from forward or downward facing cameras	15-30 fps
Onboard ultrasound	Height measurement	100 Hz
Kinect	Point cloud and RGB images	30 fps
Gazepoint Eye tracker	Pixel coordinate of gaze on a monitor	60 Hz
Eye tracking glasses	Image from field of view camera and image frame pixel coordinate of gaze	24 fps

Table 3.1: Measurement data generated by the different lab hardware components.

Fig. 3.4. The system is capable of providing the vehicle's 6 DOF motion in real time with a sub-millimeter accuracy [87].

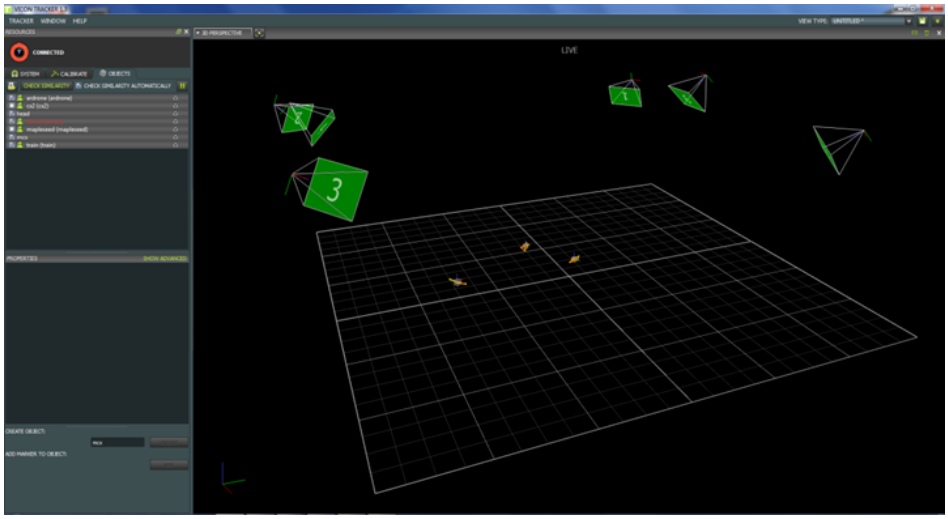


Figure 3.4: Vicon Tracker software.

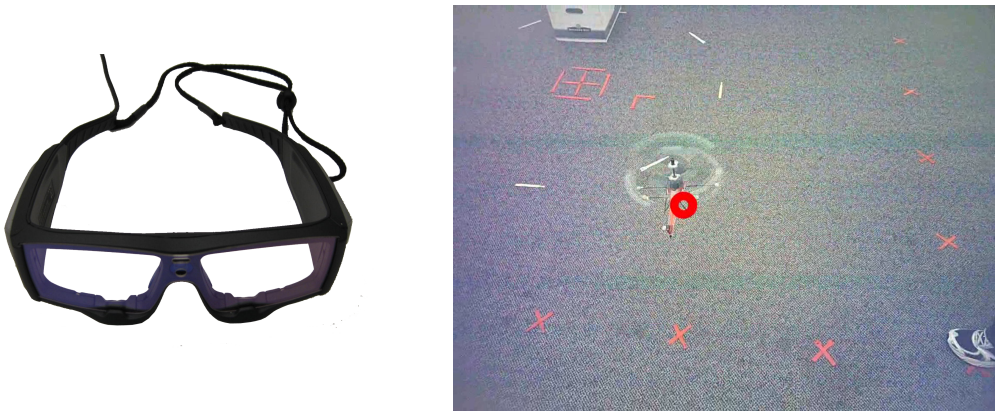
3.3.2 Operator Gaze Tracking

Tracking the operator gaze occurs in two different scenarios. In the first a pair of eye track glasses are worn while the operator to directly observe the environmental space and test vehicle. The second scenario uses an eye tracking device that attaches to a computer monitor to provide

gaze information while performing teleoperated tasks.

Gaze Tracking Glasses

Operator visual gaze tracking is achieved using a pair of Eye Tracking Glasses (ETG) from SensoMotoric Instruments SMI [88] (Fig. 3.5(a)). The ETG system consists of binocular eye tracking cameras that automatically compensate for parallax (making gaze tracking accurate at different distances). The eye tracking covers a range of 80° horizontally and 60° vertically with an accuracy of 0.5°. A forward facing scene camera (1280x960 resolution at 24 frames per second) records what the pilot observes. The scene camera has a field of view of 60° horizontal and 46° vertical.



(a) (b)

Figure 3.5: (b) SMI Eye Tracking Glasses (ETG) and (c) view from the scene camera showing the focus of visual attention as a red circle.

Data identifying the gaze location as well as the scene video are available in real-time and are integrated into the ROS environment. Integration of the ETG system allows for synchronized data from eye tracking data/video, onboard sensors/video, vehicle motion tracking, control inputs, simulated vehicles, and output from processing algorithms to be synchronized for easy analysis. Fig. 3.5(b) shows a typical view from the glass-mounted scene camera.

Data collected from the ETG system provides information necessary to model human perceptual and guidance processes. The gaze location and scene video can be combined with computer vision algorithms to extract the features or objects that are of interest to the human.

Analysis of the features in the environment can help build an understanding of what information is critical for the different pilot control modalities. The gaze location can also be used as a control input for human-machine interfacing.

Operator Display Gaze Tracking

During teleoperation experiments the pilot watches a video stream displayed on a 24-inch computer monitor. To capture the gaze location on the display, a GP3 Desktop Eye Tracker from Gazepoint is used. The device can measure gaze location at 60 Hz with an accuracy of 1 degree or less of visual angle. The system has a software API for integration with the lab infrastructure. Fig. 3.6 shows the device when positioned below the teleoperation display. In later experiments the Gazpoint device was replaced with a Tobii EyeX eye tracking device that provides similar functionality.

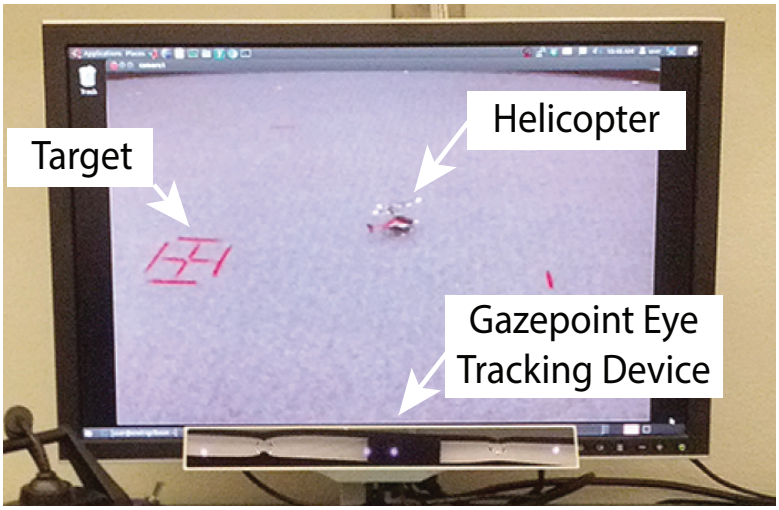


Figure 3.6: Eye tracking for teleoperation experiments.

3.3.3 Kinect Sensor

The Kinect sensor provides depth information and traditional imaging. The depth data creates a 3D point cloud that is complementary to the Vicon tracking data. The Vicon system can be used to put all the data sources in the same reference frame (vehicles, operator's head, eye tracking vector, and kinect point cloud data) and provides pose information about objects in the

experimental space. While the Vicon measurements provide very accurate information about objects, the data is sparse. The 3D point cloud from the Kinect gives dense information about the space. The Kinect point cloud data can be used to construct a 3D occupancy grid map using techniques such as Octomap [89]. The occupancy grid can then be utilized for determining the surface intersected by the gaze vector. This gives gaze information as 3D points that represent the focus of attention in the environment.

3.3.4 Actuated Camera System

For teleoperation tasks a camera is actuated using a 3-axis brushless ARRIS CM3000 gimbal. The basestation computer generates the the roll, pitch, and yaw control signals for the gimbal based on the gaze trajectory with the goal of mimicking the head portion of the head-eye system. A GoPro Hero 3+ Silver camera provides a high definition live video signal over HDMI which is captured using a Blackmagic Intensity Pro card. The captured video is displayed on the teleoperation display with a latency of approximately 120 ms. Since the latency is small and only causes a minor degradation in performance, techniques for compensating for delay are not considered further in this work. Fig. 3.7 shows the mounting, gimbal system, and camera.

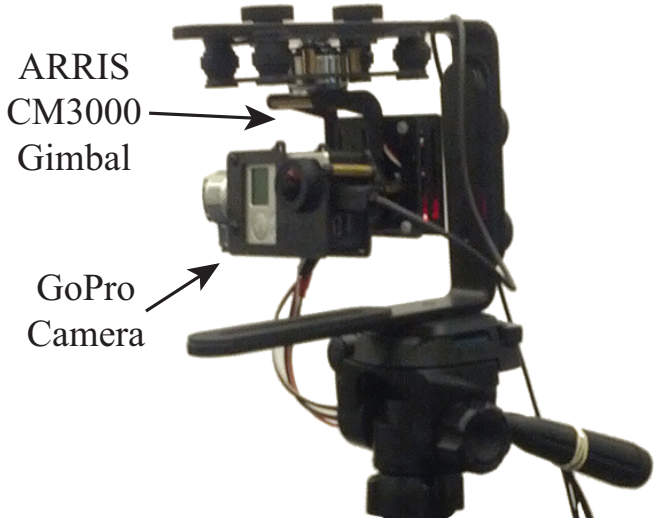


Figure 3.7: Actuated camera hardware using ARRIS CM3000 gimbal.

3.3.5 Controlled Laser Pointer

A laser pointing device was used to perform goal designation. The laser pointer was attached to the brushless gimbal shown in Fig. 3.7. The gimbal was controlled to move the laser point along a desired path on the floor. To achieve accurate positioning, the gimbal was tracked using the Vicon motion tracking system allowing for the calculation of the laser beam ground intersection. This system was used to identify and validate the head-eye system models.

3.3.6 Miniature Rotorcraft

The experimental environment described in this paper can be utilized with a variety of rotorcraft [90, 91]. The main requirements that limit the size and configuration for the rotorcraft are the ability to operate indoors and the ability to carry a sensor suite (requires a payload of at least 50 g). A range of different rotorcraft are available to conduct experiments. They include the Blade-CX2 and mCX from E-Flite [92], the Big Lama from ESky, and the AR Drone quadrotor. All are shown in Fig. 3.8. Table 3.2 provides their key physical characteristics.

	CX2	mCX	Big Lama	AR Drone
Gross w. (kg)	.23	.033	.43	.45
Payload cap. (kg)	.05	*	.15	*
Rotor diam. (m)	.35	.19	.46	.20
Length (m)	.42	.22	.46	.63
Battery (Li-Po)	2-cells	1-cells	3-cells	3-cells

Table 3.2: Physical properties of the rotorcraft.

Coaxial Platforms

The Blade CX2 helicopter is one of the primary aircraft used in the Interactive Guidance and Control Lab (IGCL). The helicopter was initially used to implement and validate autonomous guidance and control algorithms including the entire hardware and software system. Originally designed for indoor hobby flight, the helicopter's coaxial rotor system features a free floating Bell-style stabilizer bar, which is coupled to the upper rotor, and a swash-plate actuated lower rotor. Both rotors are two bladed and driven by individual electric motors. This helicopter has been the most popular commercially available miniature helicopter in this size class. In spite

of its small size and low cost it possesses all the significant functions of a full-sized helicopter. Similar capabilities are shared by the smaller Blade mCX (primary vehicle used in this work) and larger Big Lama Helicopters.

In manual mode, the helicopter is controlled through a standard RC transmitter with built-in Spektrum 2.4GHz DSM technology. Lateral and longitudinal input are implemented through a conventional swash-plate mechanism, which produces cyclic blade pitch variation on the lower rotor. Onboard electronics consist of a standard 4 channel receiver with integrated rate gyro for yaw damping feedback and dual speed controllers with mixing settings for heave and yaw motions.

Quadrotor Platform

The AR Drone shown in Fig. 3.8(d) is a commercially available quadrotor that already comes equipped with the necessary sensors. The sensors built into the AR Drone include an IMU, a set of forward and downward facing cameras, and ultrasonic sensors for height measurement. The vehicle also includes an onboard control system to stabilize the vehicle and transmit data over wifi. The video and sensor data can be integrated with custom control laws, image processing systems, or guidance laws.

(a) (b) (c) (d)

Figure 3.8: Different rotorcraft platforms used in experiments, including the: a) Blade mCX, b) Blade CX2, c) Big Lama, and d) AR Drone quadrotor.

3.3.7 Computing Hardware

The software architecture employed in the lab allows for distributed computing on heterogeneous platforms. The current configuration shown in Fig. 3.3 includes separate computer processors devoted to the Vicon motion tracking, simulation, centralized functionality such as logging and estimation, and gaze tracking. This section describes the details of the components.

Vicon Processing Computer

The Vicon processing computer runs the *Vicon Tracker* software that was described in Section 3.3.1. This computer is connected to a local area network (LAN) using a Gigabit network card. The Vicon Tracker software streams the tracking data with a very small latency (order of 10 msec) to the base-station computer through TCP/IP. The tracking data include position of the reflective markers as well as position and orientation of the centroid of the bounding box of the reflective markers. Our current Vicon processing desktop computer has an Intel core i7 processor, 8 GB RAM and running Windows 7 professional.

Base-station Computer

The base station computer is the main processing unit for real-time experiments in the lab. It hosts the operator interface, gathers the navigation data from Vicon processing computer, receives measurements from on-board sensors, captures gaze tracking data, and controls the helicopters with commands sent using a custom RC transmitter or Ethernet channel. Our current base-station desktop computer has a 2.4GHz Intel Quad Core 2 processor, 3.2GB of RAM, runs Ubuntu Linux LTS and is connected to the LAN. The user interacts with the base-station through a USB joystick and keyboard interfaces.

Gaze Tracking Computer

To interface the SMI eye tracking glasses a laptop computer with a core i7 CPU was used. The software for running the SMI eye tracking device does not provide an option for Linux operation, so a separate PC running Windows 7 was required. The gaze tracking PC utilized a custom program based on the C++ API provided by SMC to collect data and video from the glasses. The data was then sent to the basestation computer using ROS messaging capabilities.

On-board Processing Computer

The off the shelf Blade CX2 and Big Lama helicopters were customized to include on-board processing and sensors. The on-board computer is a Gumstix Verdex [91] board which runs a scaled down version of Linux. It has on-board memory and wifi communication. The components were selected to meet the payload limits and at the same time provide the functionality required for remote teleoperation, state estimation, and control. The on-board computer

is connected to the LAN and communicates with the base station through 802.11g wireless connection. Detailed discussion of on-board hardware platform are provided in [91].

Simulation Computer

The simulation computer is the main processing unit for the high fidelity simulation environment. It is used to simulate guidance and control algorithms in realistic 3D graphical environments. The simulation computer can also communicate with the base-station and Vicon computers to visualize the vehicle while flying in the lab or emulate virtual environments. The user interacts with the simulation computer through the USB joystick and keyboard interfaces. Our current desktop simulation computer has an Intel core i7 processor, 4 GB RAM, and runs Ubuntu Linux 10.04 LTS.

3.4 Software Architecture

The real-time implementation utilizes the ROS architecture to handle the communication in a multi-node application. This section describes the structure and functionality of the software components.

The base-station software is written in C++ and integrated with the motion capture system and other modules shown in Fig. 3.9 using ROS [85]. ROS is an open-source meta-operating system. It provides useful tools and code libraries including message-passing services between processes, package management and low-level device control. ROS is developed and supported by Willow Garage and widely used in the robotics community.

The real-time software implementation utilizes the ROS architecture to handle the communication in a multi-node application. For more details on the hardware and software refer to [90, 91]. Table 3.3 provides a list of the ROS modules along with the data rates and purpose of the module.

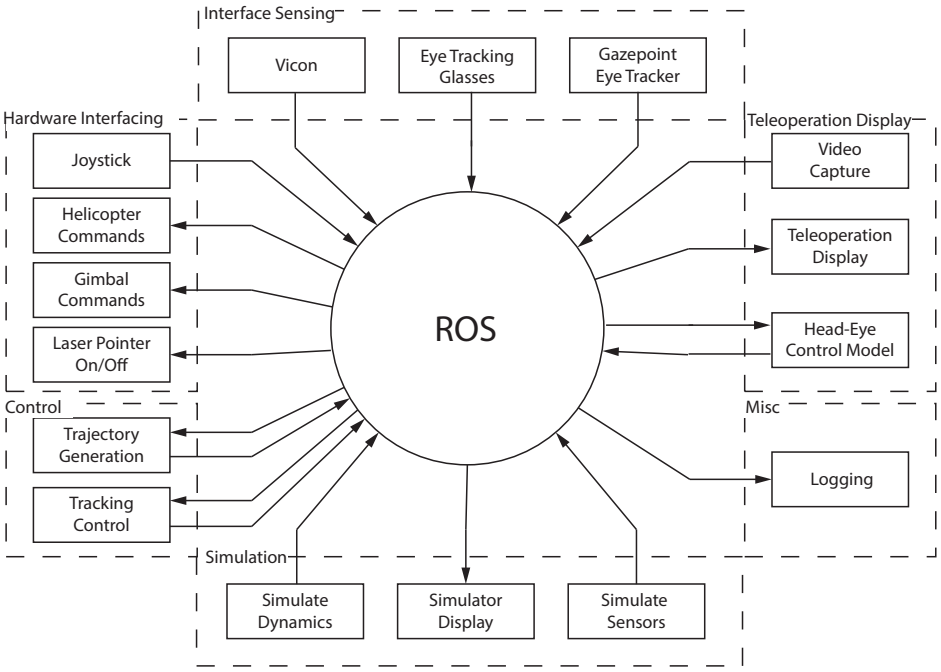


Figure 3.9: Software architecture of the system integrated around the ROS core.

ROS Node	**Update Rate**	**Purpose**
Vicon	100 Hz	Vicon Measured State
Eye Tracking Glasses	30 Hz	Gaze Position in Environment
Gazepoint Eye Tracker	60 Hz	Gaze Position on Monitor
Video Capture	30 fps	Image from Actuated Camera
Teleoperation Display	30 fps	Video Display for Operator
Head-Eye Control Model	100 Hz	Estimates and Controls Camera Orientation
Logging	100 Hz	Synchronize and Log Data
Joystick	100 Hz	Joystick
Helicopter Commands	50 Hz	Send Commands to Helicopter
Gimbal Commands	50 Hz	Send Commands to Gimbal
Trajectory Generation	50 Hz	Generate Commands for Reference Trajectory
Tracking Control	50 Hz	Control Commands
Simulator Nodes	100 Hz	Simulate Dynamics and Sensors

Table 3.3: ROS Nodes

Chapter 4

System Identification

4.1 Introduction

The primary approach for understanding the human-vehicle interactions is to use formal modeling and analysis techniques from dynamic system identification and controls. For the lab environment described in Chapter 3, the dynamic hardware elements that need to be modeled are the helicopter and the actuated camera gimbal which are highlighted in Fig. 4.1. This chapter provides the details on the system identification method and the resulting dynamic models.

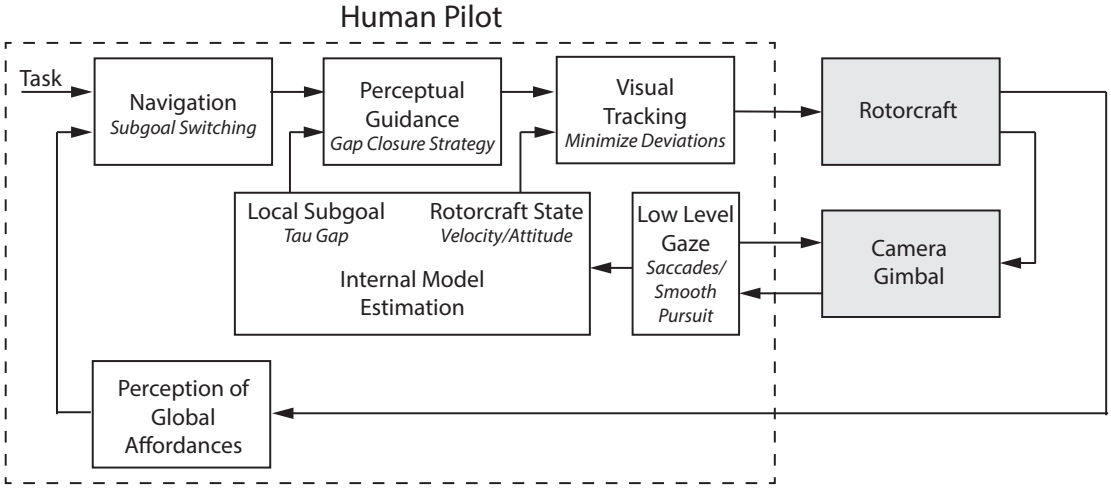


Figure 4.1: Model structure with the components to be identified highlighted.

4.2 System Identification Method

System identification techniques typically fall into two categories: time-domain methods and frequency-domain methods. In time-domain methods the numerical integration of the dynamics can cause errors in the model output due to biases in the data. Additionally, the measurement data needs to be filtered to limit the frequencies to those the model is designed to capture. Frequency-domain methods provide an alternate approach that can deal with the challenges of the time-domain methods. In particular, frequency-domain methods are well suited for system identification of aerial vehicles as discussed in [93].

The models in this work were identified by first extracting the frequency responses for measurement data and then fitting these with transfer functions using the FRESPID and NAVFIT tools in CIFER (Comprehensive Identification from FrEquency Responses) respectively [94]. Modeling data was collected using the lab infrastructure described in Chapter 3. Experiments were conducted where a frequency sweep control excitation was applied to one control input while holding the remaining control inputs for the system constant. The input-output data for each experiment was recorded at 100 Hz.

4.3 Blade mCX

The lab facility allows experiments to be conducted with a variety of miniature rotorcraft (see Section 3.3.6). The experimental work in the following chapters utilized the Blade mCX, shown in Fig. 4.2, as the main research vehicle. This section describes the identification of a dynamic model of the Blade mCX using frequency-domain techniques. Dynamic models for the Blade CX and ARDrone can be found in Appendix A. A more advanced system identification procedure for a coaxial miniature rotorcraft slightly larger than the mCX was described in [87]. For a more in depth discussion on the challenges and methods for system identification of helicopters see [95] and [96].

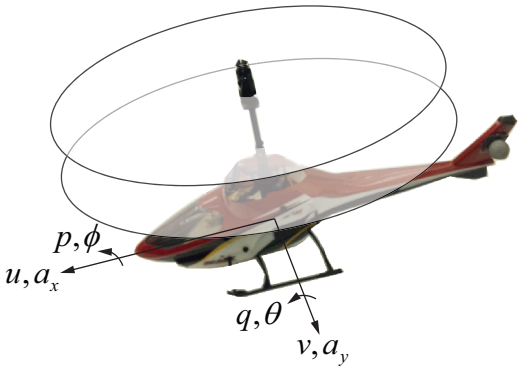


Figure 4.2: Blade mCX with the associated reference frame.

The model for the mCX can be decomposed into four parts based on the four control inputs and the resulting responses. In this work the altitude and orientation are often held constant by a human pilot for simple maneuvers and consequently the identification details are not provided. The important control dimensions are the lateral and longitudinal inputs. The basic structure of the vehicle to a longitudinal control input is shown in Fig. 4.3 [70, 4]. When a longitudinal control input δ_{lon} is applied the helicopter pitches at the angular rate q. The angular rate is integrated to produce the pitch angle θ that combined with gravity results in a forward acceleration a_x. The forward acceleration is integrated giving velocity u. The lateral control produces a similar response.

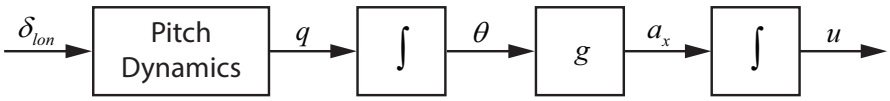
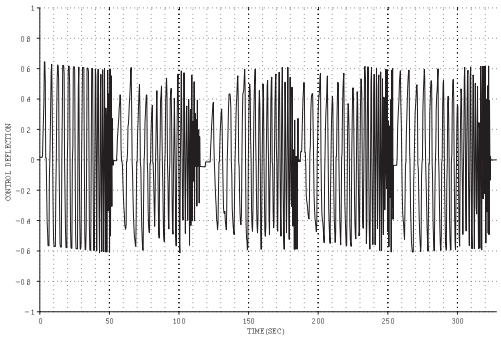
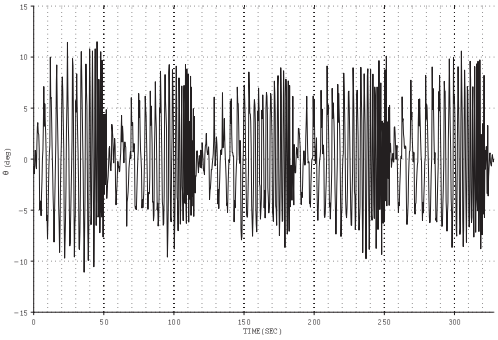


Figure 4.3: Longitudinal dynamics.

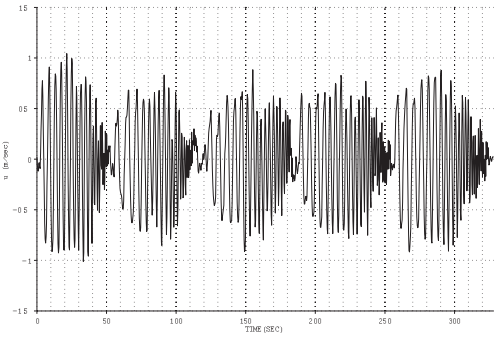
A flight experiment to collect data suitable for identifying the model shown in Fig. 4.3 was conducted. In the experiment a pilot performed 5 frequency sweeps of the δ_{lon} while holding the control inputs as constant as possible. The helicopter was returned to a stable hover between each sweep. The time histories for the experiment is given in Fig. 4.4. The figure shows the control input δ_{lon} as well as the output variables θ and u.



(a) Frequency Sweep for Input δ_{lon}



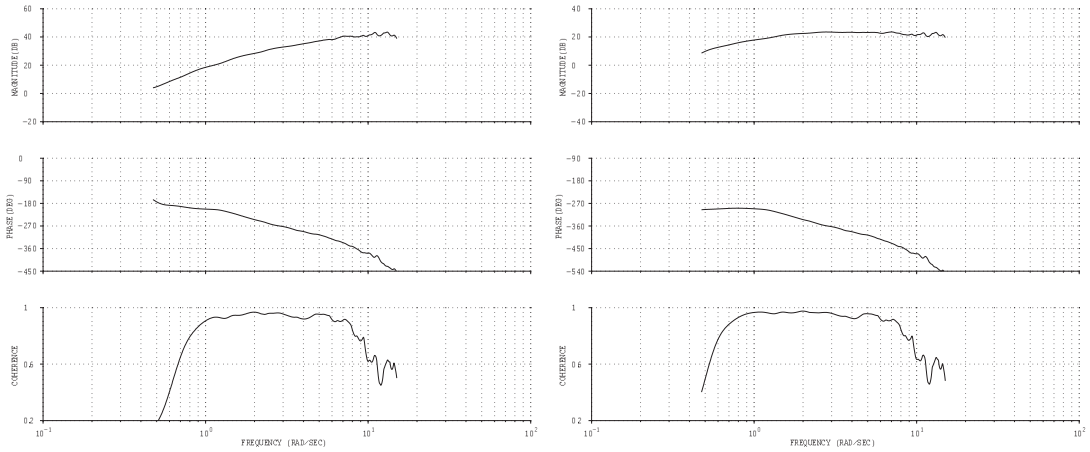
(b) Resulting pitch angle θ



(c) Forward velocity u

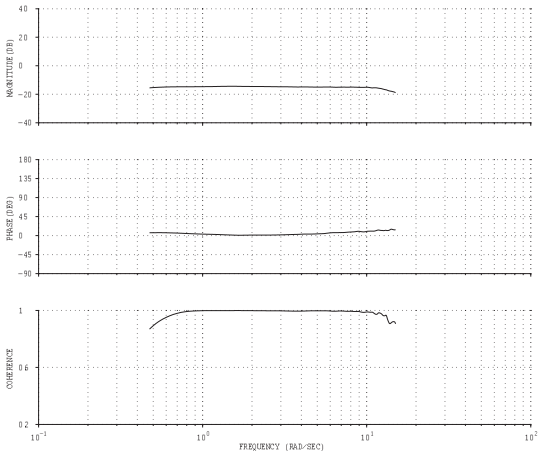
Figure 4.4: Time histories for frequency sweep experiment in the longitudinal direction.

The measured data for the longitudinal control δ_{lon}, pitch rate q, pitch angle θ, acceleration a_x, and output velocity u was imported into CIFER to extract the transfer functions between the variables. Fig. 4.5 shows the resulting transfer functions $G_{\delta_{lon}q}$, $G_{\delta_{lon}\theta}$, $G_{\theta a_x}$, and $G_{\theta u}$ are along with the coherence for each. Note that transfer functions for $G_{q\theta}$ and $G_{\theta a_x}$ are not generated due to their simple integrating nature. The transfer functions show a high coherence in the frequency range from 0.7 rad/s to 10 rad/s which corresponds to the typical human control range.

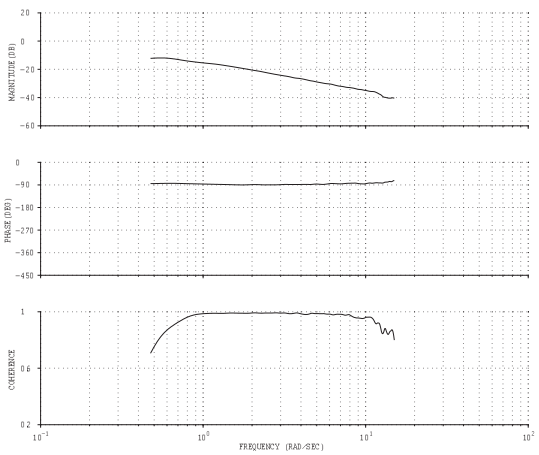


(a) Transfer function $G_{\delta_{lon}q}$

(b) Transfer function $G_{\delta_{lon}\theta}$



(c) Transfer function $G_{\theta a_x}$



(d) Transfer function $G_{\theta u}$

Figure 4.5: Longitudinal transfer functions with coherence. The top plot in each sub-figure shows the magnitude of the frequency response, the middle plot the phase, and the lower plot the coherence.

4.3.1 Attitude Dynamics

The pitch and roll in a miniature coaxial helicopter form a coupled 2-input/2-output MIMO system. The block diagram for the coupled attitude dynamics is shown in Fig. 4.6. Separate frequency sweep experiments are conducted for each input in the MIMO system. The input that is excited during an experiment is called the primary input while the second input is only used

to keep the vehicle near the center of the lab space. In the case of a longitudinal sweep the pitch θ is the on-axis response with transfer function $G_{\delta_{lon}\theta}$ and roll ϕ is the off-axis response with transfer function $G_{\delta_{lon}\phi}$.

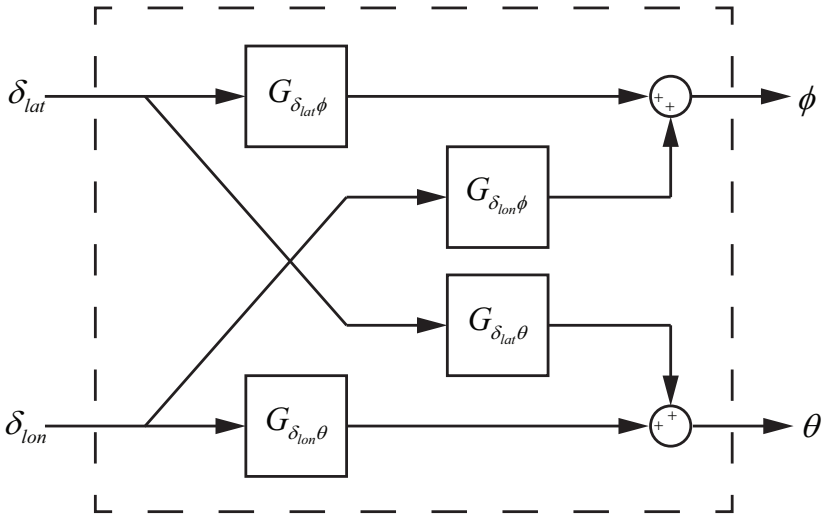
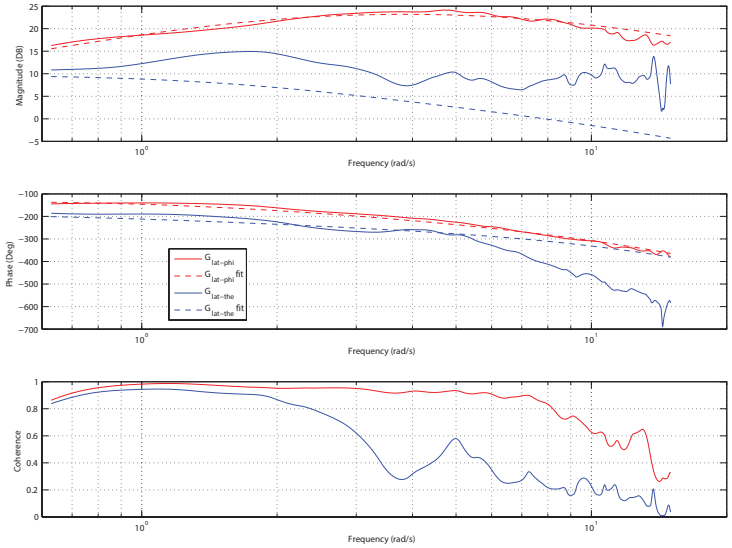
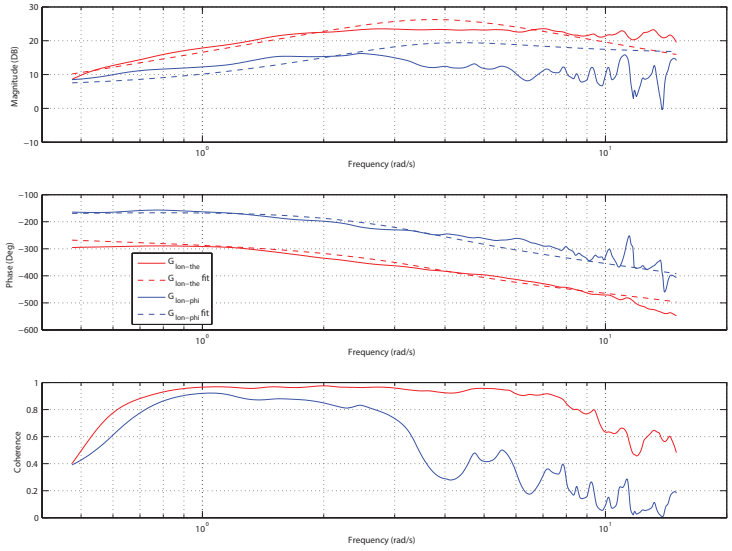


Figure 4.6: Attitude dynamics.

The frequency domain analysis that extracts both the on-axis response and off-axis response for each input sweep is a non-parametric method. The transfer functions extracted characterize the system and can be utilized to fit a parameterized model. The transfer functions for the attitude dynamics are shown in Fig. 4.7 along with their coherence plots and parameterized model fits. The figure shows the on-axis models fit the transfer function quite well with the less important off-axis providing a reasonable representation.



(a) Transfer functions $G_{\delta_{lat}\phi}$ and $G_{\delta_{lat}\theta}$



(b) Transfer function $G_{\delta_{lon}\theta}$ and $G_{\delta_{lon}\phi}$

Figure 4.7: Attitude transfer functions with coherence.

The parameterized model for each transfer function take the form of simple Laplace equations where the parameters in the numberator and denomintor are identified using the NAVFIT function in CIFER. The equations for the attitude dynamic models are given in Eqs. 4.1-4.3.

$$G_{\delta_{lat}\phi} = \frac{-143.443s - 30.515}{s^2 + 9.955s + 15.001} e^{-0.15s} \tag{4.1}$$

$$G_{\delta_{lat}\theta} = \frac{-10.0s^2 - 46.366s}{s^2 + 9.955s + 15.001} e^{-0.15s} \tag{4.2}$$

$$G_{\delta_{lon}\theta} = \frac{95.592s - 9.973}{s^2 + 4.5s + 14.028} e^{-0.075s} \tag{4.3}$$

$$G_{\delta_{lon}\phi} = \frac{6.460s^2 - 25.903s - 29.479}{s^2 + 4.5s + 14.028} e^{-0.075s} \tag{4.4}$$

4.3.2 Translational Dynamics

Similar to the attitude dynamics, the translational dynamics need to be treated as a coupled MIMO system. The translational system captures the dynamic response in the lateral or longitudinal directions to the roll and pitch outputs of the attitude dynamics. The block diagram for the translational dynamics is shown in Fig. 4.8. Using the same experimental frequency sweep data the inputs to the translational dynamics are the measured roll ψ and pitch θ angles with the measured longitudinal velocity u and lateral velocity v as outputs.

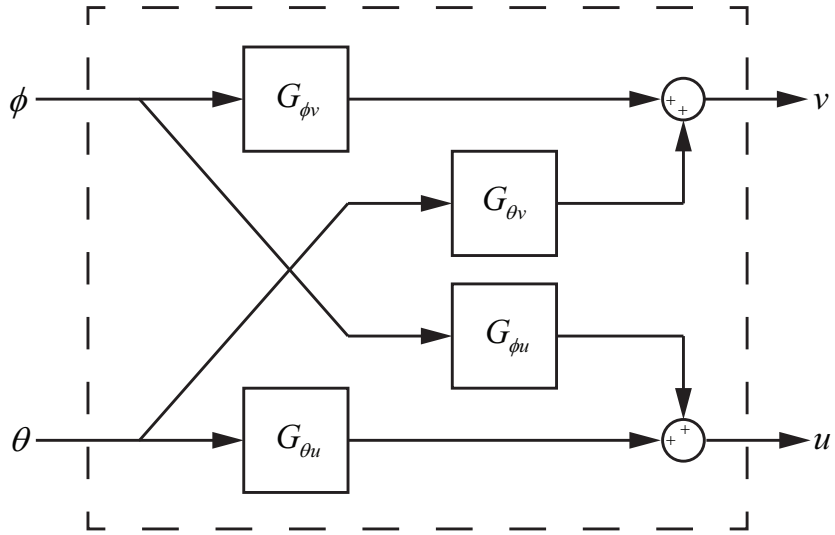
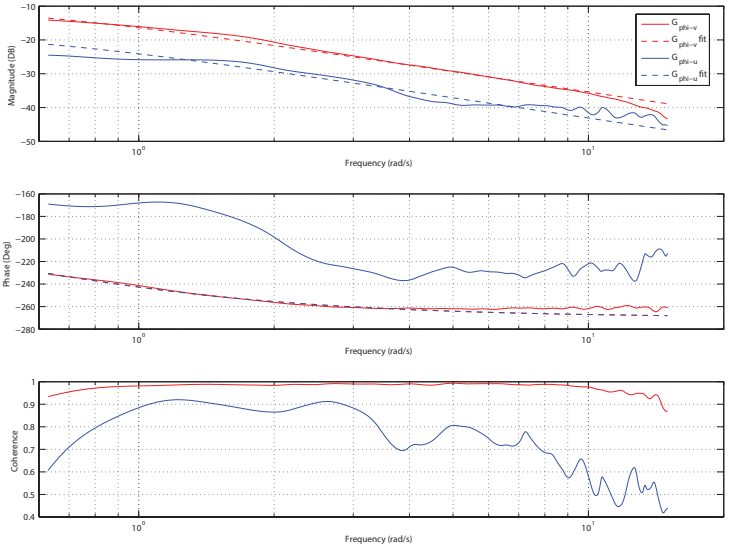
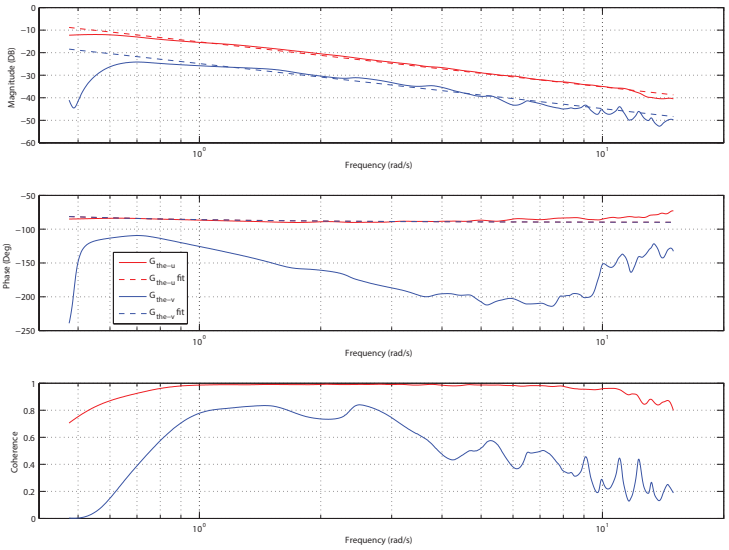


Figure 4.8: Translational dynamics.

The transfer functions for the translational dynamics are shown in Fig. 4.9 along with their coherence plots and model fits. The results show a good fit with high coherence for the on-axis responses. The off-axis response also provides a good fit with a coherence that is lower (though still more than good enough for the purposes of the model).



(a) Transfer function $G_{\phi v}$ and $G_{\phi u}$



(b) Transfer function $G_{\theta u}$ and $G_{\theta v}$

Figure 4.9: Translational transfer functions with coherence.

The model parameters are identified using NAVFIT with the results taking the form of first order transfer functions. The equations for the translational dynamic models are given in Eqs. 4.5-4.8.

$$G_{\phi v} = \frac{-0.171}{s+0.516} \tag{4.5}$$

$$G_{\phi u} = \frac{-0.07}{s+0.516} \tag{4.6}$$

$$G_{\theta u} = \frac{0.173}{s+0.072} \tag{4.7}$$

$$G_{\theta v} = \frac{0.057}{s+0.072} \tag{4.8}$$

4.3.3 Time Response

To verify the accuracy of the identified parameterized models a set of experiments was conducted where a step input was applied to one of the control inputs multiple times. The measured results show a distribution of waveforms for the roll ϕ, pitch θ, lateral velocity v, and longitudinal velocity u. Each distribution forms an envelope that can be compared to the predicted step response from the corresponding model. The model response for 5 step inputs applied to the lateral control signal is shown Fig. 4.10. The figure shows that the time response of the physical system compares well with the model predictions.

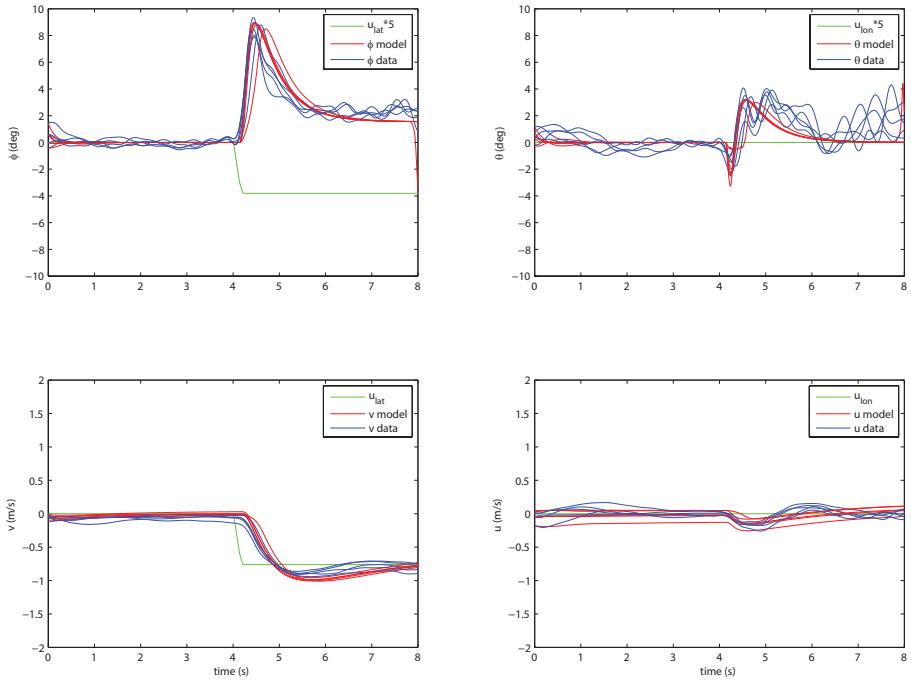


Figure 4.10: Step inputs for the lateral control signal.

The model response for 5 step inputs applied to the longitudinal control signal is shown Fig. 4.11. The model responses look like a reasonable match with the measured response.

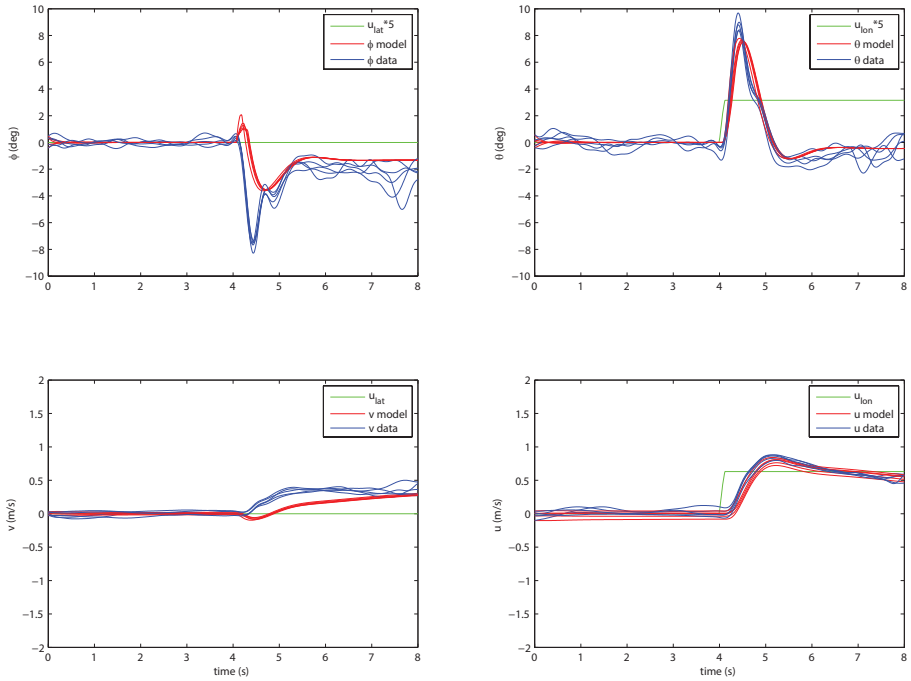


Figure 4.11: Step inputs for the longitudinal control signal.

4.4 Actuated Gimbal

The second hardware component with a dynamic response is the actuate gimbal used to rotate a camera during teleoperation experiments. The actuated gimbal is shown in Fig. 4.12 performs yaw ψ and pitch θ rotations to aim the camera at the desired target. A model of the dynamic response of the gimbal system is needed to build a control strategy that mimics the motion of an operators head in (see Chapter 8).

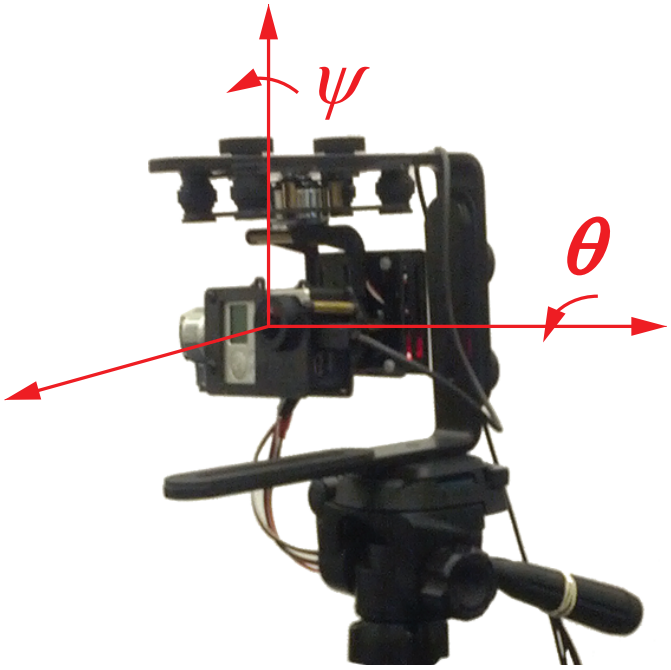


Figure 4.12: Actuated camera gimbal with reference frame.

A frequency sweep input was applied to both the yaw and pitch controls during separate experiments. The resulting dynamic responses for each input were very similar, likely due to the same type of motors providing the rotation for each axis. Due to the similarity only the system identification for yaw angle will be shown. The block diagram for the transfer function $G_{\delta_{yaw}\psi}$ from the yaw control δ_{yaw} to the yaw angle ψ is shown in Fig. 4.13.

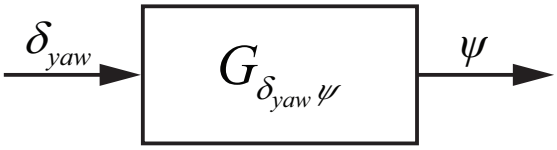


Figure 4.13: Gimbal yaw dynamics.

The resulting transfer function and parametric model fit are shown in Fig. 4.14. The frequency response and model fit are nearly identical with a high coherence over the range of interest.

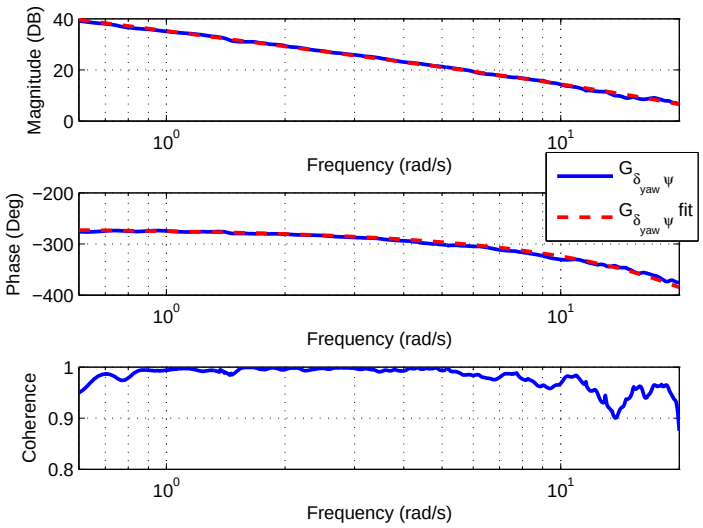


Figure 4.14: Gimbal yaw dynamics transfer functions with coherence.

The parameterized model for the gimbal yaw dynamics transfer function was identified using the NAVFIT tool in CIFER. The equations for the gimbal yaw dynamic model is given in Eqs. 4.9.

$$G_{\delta_{yaw}\psi} = \frac{1253.3}{s^2 - 21.7s - 0.065} e^{-0.14s} \tag{4.9}$$

Chapter 5

Flight Experiments

5.1 Introduction

The objective of the experiments described in this chapter is to provide data for identifying the components of the human operator's control response and perception. Fig. 5.1 shows the model blocks being investigated by the flight experiments described in this chapter. Three experimental tasks are investigated. The tasks each emphasize different aspects of the human perception-action modeling problem. The first task, hovering over a designated target, exercises the human operator's capability to reject disturbances using feedback control. The second task, flying to a goal, isolates the low-level perceptual guidance (tau gap) and control aspects of the problem. The last task, tracking a circle, extends the low-level control model to multiple dimensions and requires perceptual mechanisms to generate a reference trajectory.

Successful operation of remotely controlled vehicles depends on the difficulty of the task and the capabilities of the human pilot. Quantifying the pilot's performance can be achieved through a number of data driven measures. The data available for analysis includes the vehicle state, control inputs, operator gaze, and tracking error for a task. Measures of human performance are often thought of in terms of "the big three" [97]:

- Speed (faster is better)
- Accuracy (smaller error is better)
- Workload (less is better).

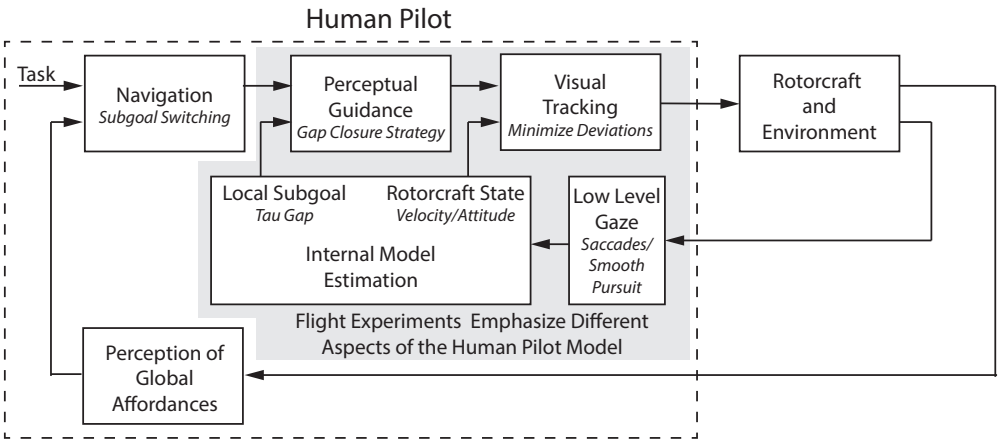


Figure 5.1: Flight experiments are designed to emphasize aspects the highlighted blocks.

In practice the measures are often a trade off with the most common example being higher speed leads to larger errors. When comparing different visual and control configurations the preferred configuration is the one that allows the most rapid completion of a task, results in the minimal error, and requires less attention.

5.2 Experimental Tasks

This section describes the three experimental tasks investigated in this work. The tasks each emphasize different aspects of humans interaction with the test vehicle and environment.

5.2.1 Hovering

The experiment in Fig. 5.2 involves the human pilot maintaining a stationary hover. This task investigates the human pilot's perceptual mechanism during simple regulation. The helicopter maintain a stable hover over a target location marked on the ground. The pilot makes adjustments as needed to keep the position as consistent as possible.

This task formulation isolates the perceptual mechanism used by the pilot during simple regulation. The hypothesis is that the human pilot focuses attention on the helicopter and uses flow information to maintain a stable hover. A second hypothesis is that the pilot will use saccade motions of the eyes to measure the distance to the target.

The pilot is instructed to start from a stationary hover over a target marked on the ground.

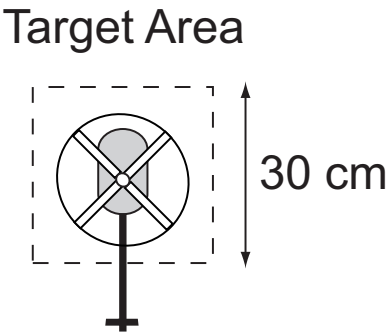


Figure 5.2: Proposed experiment for hovering.

For a given length of the time the pilot will minimize the longitudinal and lateral position error to achieve a stable hover. The experiment shown in Fig. 5.2 will be conducted for 60 seconds for each trial.

5.2.2 Target Interception

The experiment in Fig. 5.3 involves the human pilot flying directly to a target. This task exercises the human's low-level tracking control and the perceptual process for extracting the feedback control signal. The control algorithm input signal is the tracking error represented as a τ variable for the gap between the vehicle location in the image plane and the gaze location. The output of the control algorithm is the control signal corresponding to the dimension of movement. The parameters for the model can be fit to match the describing function in the frequency domain.

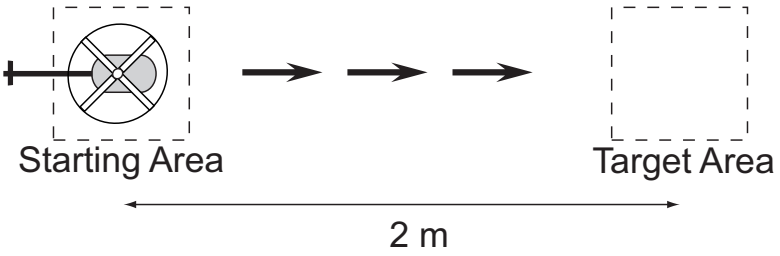


Figure 5.3: Proposed experiment for intercepting a designated goal location.

This task isolates the human's low-level tracking control and the perceptual mechanism for generating a tracking error. The purpose of the task is to isolate the basic building blocks

for modeling human perception-action. These blocks include the structure of the single loop feedback control action, the representation of the feedback error signal, and how the reference trajectory is generated (tau guide). The significance of the task is that it isolates the basic blocks that are necessary for understanding more complex behaviors.

The first hypothesis is that tau theory, developed by Lee [40], provides a method for reference generation that is applied through a feedforward element to generate an open loop response that maneuvers the vehicle from the start to the target. The second hypothesis is that existing research on human pilot modeling, based on linear control, can provide a reasonable algorithm for the model of the human feedback control action. Work by McRuer [58] and Hess [61] have shown that a linear model structure can capture the human control characteristics. The third hypothesis is that the gaze provides information on the reference trajectory.

The pilot is instructed to start from a stationary hover over the starting area and then maneuver the vehicle to the target area at one of three speeds: slow, medium, and fast speed. The experiment shown in Fig. 5.3 was conducted 10 times at each of the different speeds. For each speed, the operator is instructed to keep the velocity as consistent as possible between each of the 10 runs.

5.2.3 Circle Tracking

Fig. 5.4 shows a task where the human pilot tracks a circular trajectory. The objective for this task is to exercise the human's perceptual and control processes in multiple dimensions. The visual cues differ from the simple stationary goal described in Subsection 5.2.2 and require the human to generate an internal reference trajectory. The task also captures more complex dynamic relationships between the perceptual mechanisms and control actions when performing a curvilinear control action. The significance of the task is that it provides insight into the characteristics of the human control functions for higher dimensional tracking tasks.

The primary hypothesis for the task is that the control characteristics of the human pilot can be analyzed based on a curvilinear motion model [98]. The circle tracking task provides a curvilinear trajectory suitable for analyzing the closed-loop transfer function and loop-gain.

The pilot is instructed to start from a stationary hover over the starting area and then proceed to track a circular path marked on the ground. The pilot will perform 5 complete circles for each experiment and the experiment will be conducted 5 times each at different combinations of speed and circle radius. Using frequency domain system identification the closed-loop transfer

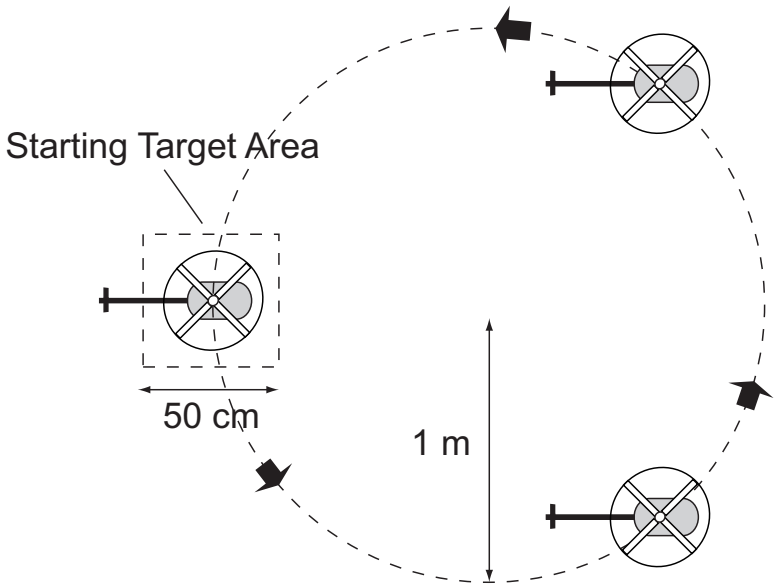


Figure 5.4: Proposed experiment tracking a circle.

function and loop-gain function will be identified for each pilot/speed/radius combination.

The result of this experiment will show how the closed-loop transfer function and loop-gain can be used to characterize the human control capabilities for more complex low-level control tasks. The control analysis will show how the control gain is adapted to provide adequate tracking performance under different operating conditions.

5.3 Characterization Methods

When evaluating different control, sensing, and display systems it is necessary to be able to measure changes in performance and operator workload for a defined set of tasks. This section describes the metrics utilized. The metrics can be broken into three categories: speed, accuracy, and operator workload. The definition of these metrics is given below.

5.3.1 Speed

For the target interception experiments the speed is calculated based on the time T to maneuver from the starting position to the target. Since the distance is constant the velocity is $v = 2/T$ m/s. In the circle tracking task T is the time to complete one cycle around the circle giving

$v = 2\pi/T$ m/s. Speed measurement is not relevant for the hover task since the objective is keep the helicopter stationary.

5.3.2 Accuracy

Tracking Error

Accuracy can be measured based on how closely the vehicle follows the desired trajectory in the xy plane. The tracking error for each task is an integration of deviation from the desired path over a single trial run. The error at a specific time is the distance between the helicopter and the closest point on the task trajectory.

Fitts' Law

Performance can evaluated based on the speed-accuracy trade-off. In 1954 Fitts published research on the relationship between speed, movement distance, and accuracy. The approach utilized a "Fitts Task" where an object was moved along a linear path between two locations [99]. The relationship is given by

$$Movement\ Time = a + b\log_2\left(\frac{2D}{W}\right) \tag{5.1}$$

where A is the distance between locations, W is the target width, and the linear relationship (constants a and b) is empirically identified. A revised relationship was proposed by Schmidt [100]. Schmidt found that for tasks requiring a single aimed movement, a linear relationship existed between the movement speed and the effective target width. The revised relationship is

$$W_e = a + b(A/Movement\ Time) \tag{5.2}$$

where W_e is the effective target width. To calculate W_e the within-trial error was measured to find the standard deviation or "spread" around the target locations. The effective width, W_e, was defined as the width necessary to capture 96% of the identified distribution.

5.3.3 Workload

An important measure for assessing a helicopter system for a task is how much workload is required by the operator to successfully perform the task. The challenge in objectively and

quantitatively addressing this question is that the both the task and human control include measurable quantities such as control signal variation as well as subjective measures of the task difficulty [101]. Two metrics based on the control signal are defined.

Attention Metric

The first workload metric is a measure defined by the attention functional in [102]. Brockett defined an attention metric based on the control signal u, state x, and time t. The original metric had two terms $\left(\left\|\frac{\partial u_i}{\partial t}\right\|\right)^2$ and $\left(\left\|\frac{\partial u_i}{\partial x_i}\right\|\right)^2$. In this work we focus on the first term which looks at how the control signal changes versus time. The resulting metric is defined as:

$$\eta = \frac{\sum_i \left(\left\|\frac{\partial u_i}{\partial t}\right\|\right)^2}{T}. \tag{5.3}$$

The attention functional is intended to be a measure of how much effort the operator needs to supply in order to complete a task. In general, the amount of effort is roughly proportional to the magnitude and frequency of the control adjustments that the operator needs to make. When executing a task where the control inputs are held relatively constant (such as a stable hover), the attention functional would be small. A task that required significant control (such as navigating a slalom course) would have a much larger attention functional.

Pilot Crossover Model

Another method of measuring operator workload is to look at the pilot crossover model originally proposed by McRuer [56]. It was shown that the combination of the human pilot (Y_p) and the helicopter dynamics (Y_h) can be approximated by the following equation near the crossover frequency ω_c [57]:

$$Y_p Y_h = \frac{\omega_c e^{-\tau s}}{s}. \tag{5.4}$$

The technique used to determine the cutoff frequency was described in [103]. Using this approach, the cutoff frequency is defined as the half power point of equation (5.5). In the equation, $G_{\delta\delta}$ is the auto-PSD function for the control inputs. The cutoff frequency provides an indication of how much effort an operator exerting.

$$\sigma^2 = \frac{1}{\pi}\int G_{\delta\delta}\, d\omega \tag{5.5}$$

5.4 Test Pilots

Six undergraduate students from the University of Minnesota performed the flight experiments in 5.2. The pilots were recruited from the aerospace department and had past experience operating RC helicopters. The test pilots demonstrated a range of skill levels. Two pilots possessed significant experience and were able to complete tasks quickly and accurately. Another two pilots had some experience operating miniature rotorcraft but performed with less accuracy than the highly skilled operators. The last two pilots in general displayed the least accuracy. The modeling work in subsequent chapters was based on the test data from the highly skilled pilots.

Flight tests were conducted in one hour sessions that occurred weekly. At the start of each session a calibration procedure was required for the eye tracking system. The pilots were then allowed to practice the task until they felt comfortable. To maintain consistency between experiments the pilots were seated at the same location for all test flights.

Chapter 6

Model of Pilot Control

6.1 Introduction

Human control requires a number of processes and mechanisms acting in concert to achieve the level of performance seen in skilled individuals. Pew identified three levels of control organization [20]. The most basic level of control generated by a human pilot is that of a simple servomechanism where motor outputs act to correct the error between the perceived and desired states. This basic control element can be represented by concepts from the theory of feedback control and provides a basis for all higher levels of control organization. The next level of control incorporates the coherence and predictability of the task and environment. The human pilot generates a desired trajectory based on patterns of behavior learned from past experience. The last level of human control draws from the environment cues that identify a goal to call up from memory and an integrated motion model that can achieved the desired result. This chapter models the two lower levels of the human pilot's control organization for simple flight tasks. The two components highlighted in Fig. 6.1 represent these two levels.

The chapter is organized as follows. The next section describes the structure of a linear control model and identifies specific models for feed-forward and feedback control. Section 6.3 discusses human control when the vehicle moves along curvilinear trajectories. Section 6.4 provides an overview of tau guides and identifies a perceptual guidance model for a step task. Lastly, Section 6.5 provides conclusions.

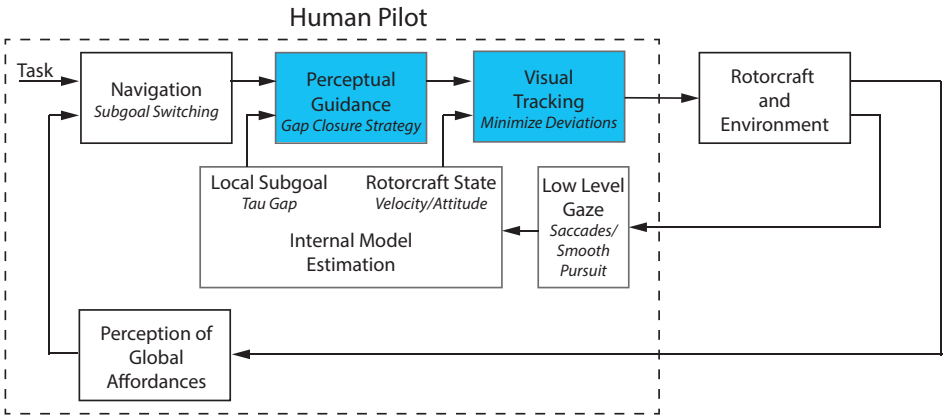


Figure 6.1: This chapter investigates human visual tracking capabilities (blue box). The block diagram shows how the different components of the overall model structure are related.

6.2 Visual Tracking: Linear Control

To operate a vehicle along straight trajectories the visual tracking component in Fig. 6.1 can be modeled using dynamic linear elements. Fig. 6.2 gives a detailed breakdown of how the visual tracking component can be implemented. In the figure the perceptual guidance block generates a reference velocity which passes through a feedforward element to generate a control signal δ_{fb} that drives the vehicle velocity to match V_{ref}. In parallel, the reference velocity is integrated and compared with the actual position to generate a positional error x_{err}. The positional error provides the input for the feedback part of the visual tracking system. The feedback element generates an error correcting control signal δ_{fb}. The control signals δ_{fb} and δ_{ff} are added to get the output of the visual tracking block δ.

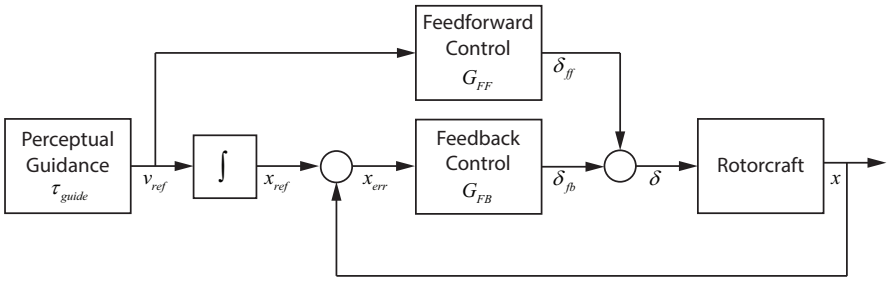


Figure 6.2: Structure of the visual tracking loop.

6.2.1 Feedforward Control

The feedforward control element takes a velocity reference and outputs a control signal that produces the desired velocity in the helicopter. In Chapter 4 a dynamic model of the miniature rotorcraft used in the flight experiments was identified. A simplified version of the model from the longitudinal control signal δ_{lon} to the vehicle's forward velocity u is given in Eq. 6.1.

$$G_{\delta_{lon}u} = \frac{2.802}{s+2.029} \tag{6.1}$$

Inverting the simplified model of $G_{\delta_{lon}u}$ gives the feedforward dynamics that map a velocity reference V_{ref} to control signal δ_{ff} that will drive the vehicle to the desired velocity. Eq. 6.2 gives the dynamic model for the feedforward block G_{FF}.

$$G_{FF} = 0.724 + 0.357s \tag{6.2}$$

6.2.2 Feedback Control

Based on the Adaptive Pilot Model described in [69] the pilot control model takes the form of Eq. 6.3. Estimation of the pilot control parameters for the feedback element can be accomplished using frequency domain estimation methods [104].

$$Y_P = K_P + K_D s \tag{6.3}$$

To estimate the control parameters K_p and K_d a forcing function is applied as a disturbance to the control signal as shown in Fig. 6.3. The forcing function applied is the sum of multiple sine waves that provide a rich excitation for the system [105, 2]. The multisine equation f_d is defined by Eq. 6.4 with the parameters given in Table 6.1.

$$f_d(t) = \sum_{i=1}^{10} A_d(i)\sin(\omega_d(i)t + \phi_d(i) \tag{6.4}$$

Data was collected for the hover task from Section 5.2.1. While the pilot performs the hover task the forcing function is injected into the control input causing continuous displacement of the vehicle. This requires the pilot to provide control actions to return the vehicle to the desired hover position. For the hover task the reference angle from Figure 6.3 is constant resulting in the error angle being directly related the visual angle. In this case measurements are available for the input ($errangle$) and output (δ) of the feedback control block Y_P. The NAVFIT function

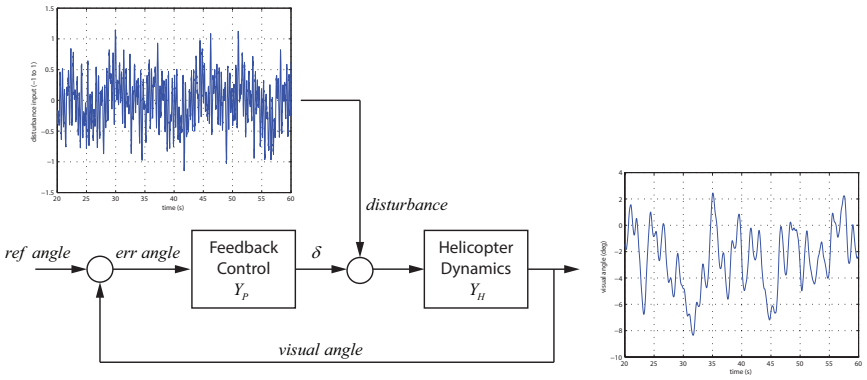


Figure 6.3: For the hover task, a known disturbance is injected into the control signal. The control parameters for Y_P are then identified.

ω_d rad/s	A_d deg	ϕ_d rad
0.383	0.048	-2.088
0.844	0.175	1.238
1.764	0.381	-3.895
2.838	0.502	3.138
3.912	0.581	-2.807
5.446	0.684	-1.808
7.747	0.866	-1.563
10.508	1.152	-2.953
13.116	1.496	-2.626
17.334	2.212	0.864

Table 6.1: Multisine forcing function coefficients [2].

of CIFER was used to identify the parameters fo the feedback control block [94]. Figure 6.4 shows the transfer function fit for Y_P along with the nonparametric frequency response extracted from the experimental data.

The transfer function fit in Figure 6.4 has a high coherence in the frequency range of interest (below 10 rad/s) and provides a reasonable match to the experimental data. The parameters identified for Y_P can be found in Eq. 6.5 with Eq. 6.6 showing G_{FB}.

$$K_P = -1.19, K_D = -0.94 \tag{6.5}$$

$$G_{FB} = -1.19 - 0.94s \tag{6.6}$$

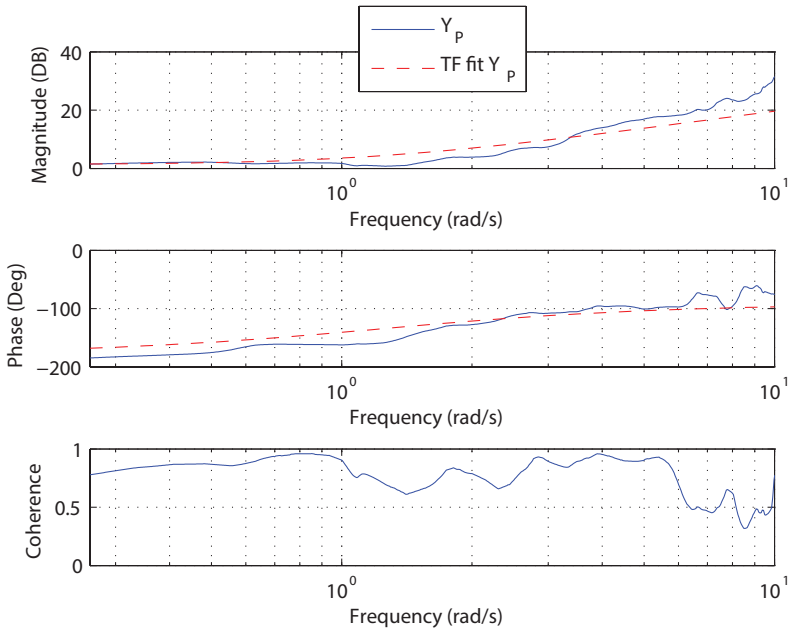


Figure 6.4: The control parameters are then identified using system identification techniques.

In [4] it was shown that if the pilot-vehicle short-term attitude dynamics (stabilizing attitude) are assumed to follow the crossover model [56], the dynamics of the free response to a displacement for the complete pilot-vehicle system can be reduced to a 2nd order form with natural frequency $\omega_n = \sqrt{gK_P}$ and damping ratio $\zeta = gK_D/2\omega_n$. Factoring in the gain from the control signal δ to the attitude angle at the approximate crossover frequency ($\omega_c = 1.0$ rad/s) gives a natural frequency and damping ratio of 1.39 and 0.55 respectively.

6.3 Visual Tracking: Curvilinear Control

6.3.1 Motion Model

Curvilinear motion is defined as an object moving along a curved path. In Figure 6.5 a rotorcraft follows a trajectory that includes a curved path. During the curved part of the trajectory the path is defined by the velocity v and the turn radius r. A constant velocity and radius a result in circular path as shown.

Most curvilinear motions involve nonlinear dynamics. The simplest physical model is that

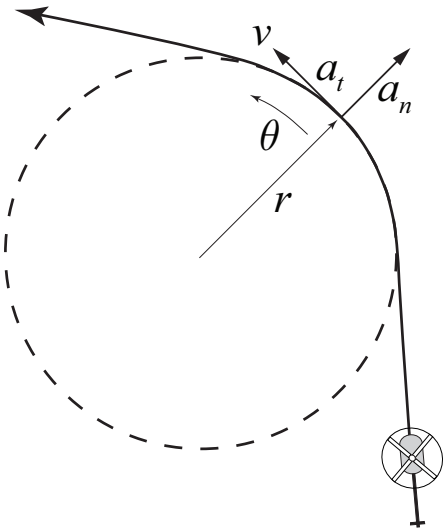


Figure 6.5: Curvilinear motion defined by a velocity v and turn radius r.

of a point-mass or particle system defined by polar coordinates to give the following model for radial acceleration:

$$\ddot{r} - \dot{\theta}^2 r = a_n \tag{6.7}$$

where $\ddot{r}$ is the inertial acceleration, $\dot{\theta}r$ is the inertial acceleration due to centrifugal effect and a_n is the external acceleration representing the control input. Different physical processes are utilized depending on the application. For an aerial vehicle, it is achieved by vectoring the lift or thrust vector. For an automobile, it is achieve by the skin friction resulting from reaction forces produced by steering the front wheels; for a bicycle, it is achieved by the reaction friction resulting from the combination of change in body posture and steering.

6.3.2 Control Characteristics

Based on the nonlinear equations of motion, at a given time the system will be at an operating point specified by the speed v and the angular rate $\dot{\theta}$, i.e., the operating point is given by the point (v_0, r_0). In general the control behavior can take place over a range of operating points.

Basic dynamic characteristics and more specifically path tracking control characteristics can be determined from the linear analysis. The linearized equation of motion with a constant $\dot{\theta}$ is

given by:

$$\ddot{\delta r} = \frac{v_0^2}{r_0^2}\delta r + \delta a_n. \tag{6.8}$$

The transfer function for the cross-track response is:

$$G_r(s) = \frac{\delta r(s)}{\delta a_n(s)} = \frac{1}{s^2 - \dot{\theta}_0^2}. \tag{6.9}$$

Notice that this system has two poles on the real axis $s = \pm\dot{\theta}_0$. One pole is stable while the other is unstable. To be stabilized, using the acceleration as control input this system requires a zero in the left half plane. This control can be realize using a a PD control element $K(s) = K_p + K_d s$. With this control system the characteristic polynomial of the closed-loop system is:

$$s^2 + K_d s + K_p - \dot{\theta}_0^2. \tag{6.10}$$

The gains K_d and K_p of the PD elements can be selected to provide appropriate transient response (damping and settling time). The settling time $T_s = 4/(\zeta\omega_n)$ for the path following should be about twice that of the attitude dynamics.

Note that the path following response characteristics depend on the operating point. In particular the natural frequency depends directly on the proportional gain K_p and operating point ${\omega_n}^2 = K_p - \frac{v_0^2}{r_0^2}$. The derivative gain K_d determines primarily the damping $K_d = 2\zeta\omega_n$, where ζ is the damping ratio. Notice that engaging in a turn while keeping the speed constant, will decrease the natural frequency of the path tracking response. Increasing the turn rate decreases the natural frequency as follows:

$$\omega_n^2 = K_p - \frac{v_0^2}{r_0^2} = K_p - \dot{\theta}^2. \tag{6.11}$$

Therefore, unless the proportional gain K_p is increased, the natural frequency will decrease. The physical interpretation is that at higher turn rate, corrections in radial position requires larger normal acceleration. Therefore, for the same perturbation in position δr a larger gain is required. Note that this effect is consistent with piloting experience and other path tracking tasks (e.g. bicycle or car). More details on the curvilinear control analysis can be found in [106].

6.3.3 Experimental Results

This sections gives a summary of experimental results conducted with a miniature helicopter. The experiments were conducted in the Interactive Guidance and Control Lab (IGCL) [84] using a miniature Blade mCX helicopter. The experiments were conducted using human subjects.

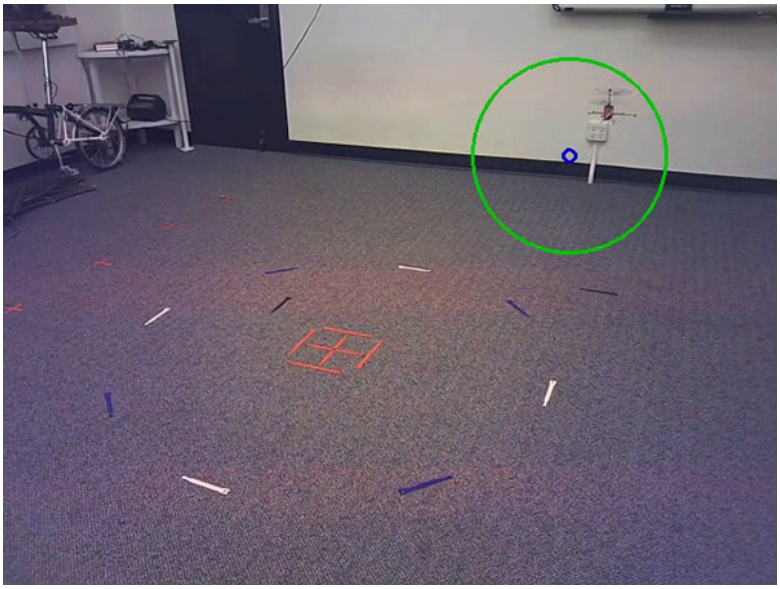


Figure 6.6: Blade mCX helicopter during flight experiments.

The goal of the experiments is to investigate the human operator's adaptation mechanisms. Human operators have been shown to be highly adaptive in a range of control tasks [56]. Therefore, some of the key questions are to what extend does the human adapt the path control feedback strategy? What criteria govern the human control strategies; does the operator use a constant gain, or does the human adapt the gain based on maintaining similar outcome? Research in motor control suggest that humans adapt the control performance based on acceptable error [107].

Method

For the experiments, a circle 1.5 m in diameter was marked on the floor in the lab as illustrated in Fig. 6.6. The subject was instructed to fly the helicopter at four operating points based on the following guidelines: The subject was asked to fly circles at two different radii and each at two speeds. For the smaller circle the subject was constrained to remain inside the dashed lines and for the larger circle, outside the dashed lines. For each the subject was to track the circular path at slow and fast speeds.

Loop Gain

The investigation of the human control performance is based on closed-loop identification. A frequency domain identification technique was applied to identify the loop gain function. During test flights the pilot follows an internal reference of the desired trajectory. In contrast to the automatic control where the path tracking controller receives a reference path, with the human subjects it is not possible to explicitly know the internal reference trajectory utilized. The center and radius of the reference circles can be estimated based on the trajectory statistics. The estimate of the internal reference can then be used to estimate the tracking error.

If we assume that the operating point corresponds to the mean radial position, then the observed closed-loop performance can be interpreted as the response of the system as it is driven by a process noise n:

$$\delta r = H_{\delta r0,\delta r} n \tag{6.12}$$

where δr is the deviation from mean radial position and $H_{\delta r_0,\delta r}$ is the closed-loop transfer function.

If we assume a process noise with a uniform power and a variance based on the actual process noise

$$n = \mathcal{N}(0,\sigma^2) \tag{6.13}$$

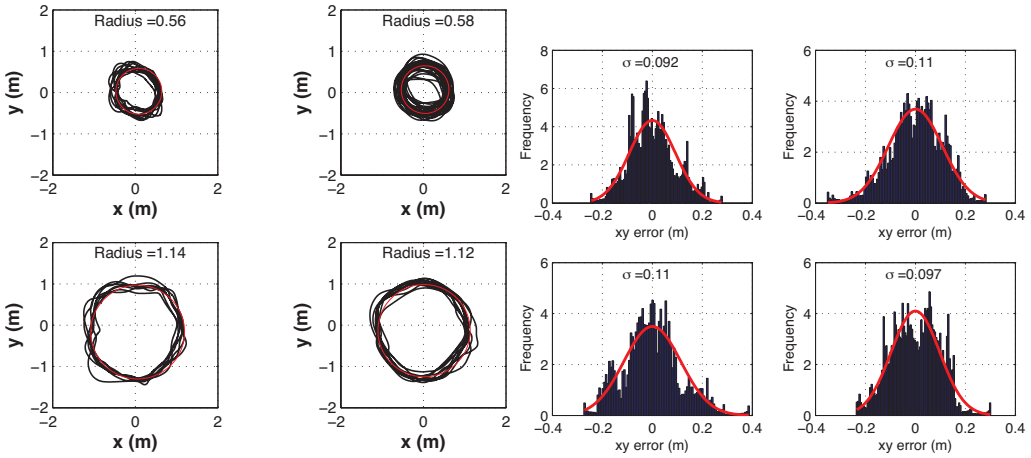
then the estimated closed-loop frequency response should be near the true response. The close-loop frequency response can be estimated from:

$$\hat{H}_{\delta r_0,\delta r} = \frac{G_{n,\delta r}}{G_{n,n}} \tag{6.14}$$

where $G_{n,\delta r}$ is the cross-spectrum for the radial position $\delta r(t)$ and the process noise n corresponds to the noise around the mean radial position for each experiment. $G_{n,n}$ is the process noise auto-spectrum. The frequency response extraction was performed using the FRESPID tools from the CIFER toolbox [108].

Fig. 6.7(a) shows the ground track for the four conditions (small circle at slow and fast speeds and a large circle at slow and fast speeds) where the small circular radius was slightly over 0.5 m and the larger radius around 1.1 m. The mean for the slow velocity conditions was found to be around 0.35 m/s while the fast speed was approximately 0.7 m/s.

Fig. 6.7(b) shows the histograms of the error magnitude for each of the circle tasks. Notice that the standard deviations for the four test cases are very similar, indicating that the human



(a) Ground track (b) Tracking error

Figure 6.7: EData for human tracking experiment for the four operating conditions. The trajectory data was used to generate the statistics for the key variables.

pilot's tracking error bound is largely independent of the operating conditions and therefore suggesting that the human adapts the path tracking gains.

Fig. 6.8(a) shows the second-order transfer functions fitted to the four frequency responses extracted from (6.14) after scaling, to give a 0dB gain at low frequency. The estimate of the loop-gain function can then be computed from

$$\hat{L}_{\delta r_0,\delta r} = \frac{\hat{H}_{\delta r_0,\delta r}}{1-\hat{H}_{\delta r_0,\delta r}} \tag{6.15}$$

The magnitude for the loop gain function for all four experiments are shown in Fig. 6.8(b).

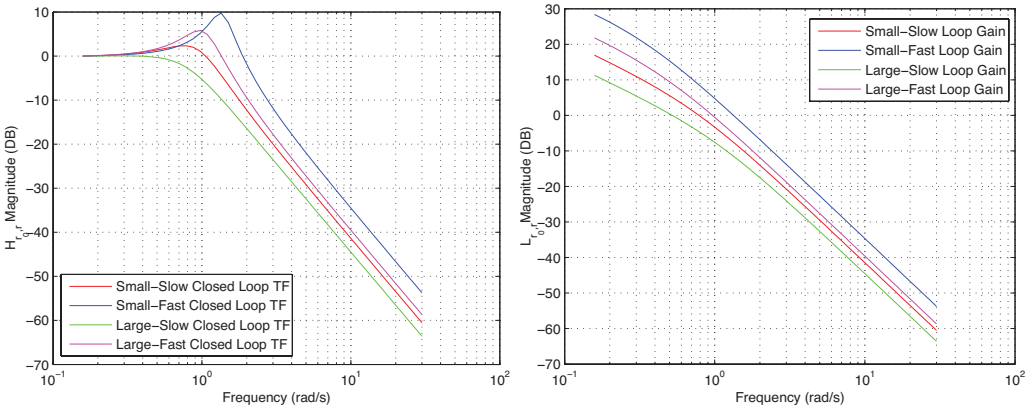
Table 6.2 gives the bandwidth, loop gain and crossover frequency for the four operating conditions. The table also shows σ_r, which provides a measure of the experimental tracking error for each trial. Data from a test pilot with an advanced skill level is shown.

The transfer and other loop gain functions show a clear increase in bandwidth with the increase in turn rate. As expected, the loop gain shows a significant increase in the loop gain magnitude (from -9.36dB for the large/slow circle to 6.14dB for the small/fast circle). The extracted transfer function captures both the effect of operating point on the plant and the human control element. The increase in the natural frequency in the closed-loop transfer function and the increase in loop gain magnitude both reflect the increase in the human path tracking gain K_p. The results show that the human subject adapts the feedback gain to achieve a similar level

Table 6.2: Identified closed-loop and loop-gain characteristics (sorted according to angular rate).

Experiment	Pilot 2 ω_c (rad/s)	BW (rad/s)	$L(j\omega = 1)$ (dB)	$\dot{\theta}$ (rad/s)	σ_r (m)
Large / slow	0.37	0.52	-9.36	0.38	0.158
Small / slow	0.83	1.40	-2.62	0.61	0.112
Large / fast	1.22	1.91	3.29	0.72	0.114
Small / fast	1.73	3.20	6.14	1.48	0.115

of tracking performance as measured by the error variance. Another insight attained by the identification is that the transfer function damping decreases with increasing bandwidth, which suggests that the operator is not as good at increasing the damping.



(a) Closed-loop transfer function

(b) Loop gain function

Figure 6.8: Frequency response for the a) closed-loop control and b) loop gain transfer function identified from experimental data based on (6.14-6.15).

Path Tracking Control Model

A deeper characterization of the control behavior requires parametric identification that captures the feedback control parameters at the operating point. The experimental results shown in Section 6.3.3 identified the closed loop transfer function for the cross track system. Using the identified transfer function the control gains (Kp and Kd) for the outer loop of the crosstrack

control system were identified using:

$$K_p = \omega_n^2 + \dot{\theta}_0, K_d = 2\varsigma\omega_n \tag{6.16}$$

Table 6.3 gives the parameters for the tracking control identified for the four different operating conditions. The increase in the natural frequency in the closed-loop transfer function corresponds with the increase in the human path tracking gain K_p. The human subject adapts the feedback gain to based on task conditions to maintain a consistent level of error.

Table 6.3: Control gains for tracking control.

Experiment	Pilot 2				
	$\dot{\theta}_0$ (rad/s)	ω_n (rad/s)	ς	K_p $(1/s)^2$	K_d (1/s)
Large / slow	0.38	0.83	0.96	0.84	1.60
Small / slow	0.61	0.78	0.15	0.99	0.24
Large / fast	0.72	1.15	0.34	1.84	0.78
Small / fast	1.48	1.64	0.44	4.89	1.46

6.4 Perceptual Guidance

Research on sensory guidance has demonstrated that guidance performance can be described using relatively simple principles. The main approach, called tau theory [40], utilizes a simple variable tau. Tau is defined as the time-to-closure of a gap at the current gap closure rate. A second principle of tau theory is tau-coupling where two or more tau gaps maintain constant ratio relative to one another. Gaps can also be closed using intrinsic guides, called tau guides, that are internally generated mental models of the desired motion [41]. When utilizing these tau guides, the externally perceived gap is coupled with the internal guide. The form of the tau guide depends on the type of motion with examples being constant acceleration, constant deceleration, or acceleration-deceleration maneuvers. The perceptual guidance block in Fig. 6.1 can be implemented to provide the reference trajectory for a given task based on the appropriate form of a tau guide. For the step task an acceleration-deceleration tau guide is required. The form of the tau guide τ_g is given in Eq. 6.17.

$$\tau_g = 0.5(t - \frac{T^2}{t}) \tag{6.17}$$

The tau guide is converted into a reference velocity by Eqs. 6.18 and 6.19 which are based on the tau coupling principle and the definition of the tau variable.

$$\tau_x = k\tau_g \tag{6.18}$$

$$v_{ref} = \frac{x}{\tau_x} \tag{6.19}$$

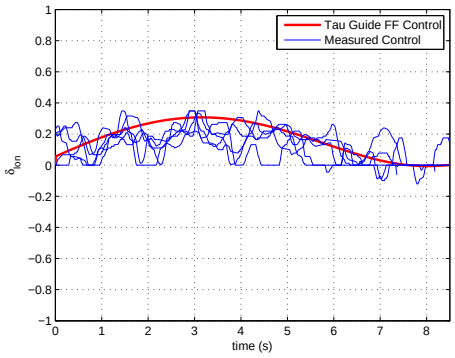
Experimental results for the step task at slow, medium, and fast speeds show that the human control response can be approximated using a tau guide. Fig. 6.9 shows the correspondence between control signals generated using a tau guide and the actual controls from the human pilot. The value of k was 0.3 and the T values (time to complete the task) for the slow, medium, and fast speeds were 2.5 s, 5.5 s, and 8.5 s respectively. The control signal for the fast speed plateaus due to joystick limits on the maximum control action.

6.5 Conclusions

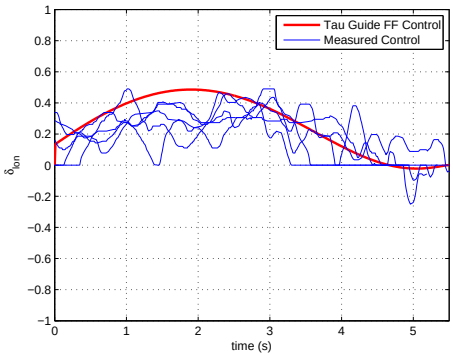
This chapter identified the visual tracking and perceptual guidance components of the proposed model of the human pilot. The visual tracking was investigated from the linear and curvilinear control perspectives. The perceptual guidance component implements the concept of a tau guide to generate a reference trajectory for a given task.

For the linear control case, feedforward and feedback elements were identified. The feedforward component generates an open loop control response that drives the vehicle along the desired trajectory for a task. The feedback element corrects errors between the desired state and the perceived state of the vehicle.

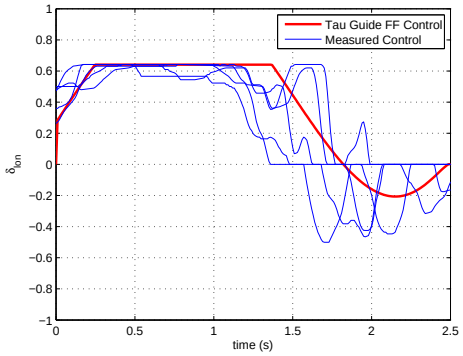
The curvilinear control section described effects of changing operating point on path tracking performance based on classic control analysis. The main insight is that the path tracking gain has to be adapted to provide adequate bandwidth and steady-state error. The results also explain the type of mechanism required to adapt to effects of changing operating point and still ensure adequate tracking performance. These insights are relevant both for design of vehicle control and assessment of human performance.



(a) Control for Step Task at Slow Speed



(b) Control for Step Task at Medium Speed



(c) Control for Step Task at Fast Speed

Figure 6.9: Augmented flight control provides a feedforward control signal to partially automate the step task. A tau guide creates a reference velocity that is converted to a helicopter control signal. The subfigures show the comparison of the generated signal against a set of control human controls.

Chapter 7

Gaze for Guidance and Control

7.1 Introduction

Teleoperation of agile robotic systems in three dimensional environments can benefit from a detailed understanding of the perceptual control mechanisms used by the operator both for the design of operator interfaces and for the use of gaze information as part of the control mechanism. The objective of this chapter is to characterize the role and contribution of the operator's gaze motion in remote control operation of an agile vehicle and show how gaze can be used to estimate control relevant information. Fig. 7.1 shows the model components discussed in this chapter. The components decode gaze and use the resulting information to estimate the state of the vehicle and environmental features.

The goal of the work presented in this chapter is to determine and model the pilot's visual perception during precision remote-control (RC) operation of a miniature rotorcraft. For a given task, the control problem involves processing visual input and transforming the result into commands to the relevant musculature. Some of the key questions include: what type of information is required for the different control modalities; and how is this information extracted and used as part of the perceptual control mechanisms. To answer these questions an approach based on the analysis of the closed-loop operator-agent-environment interactions within a control theoretic framework is used. Experiments were conducted using a miniature remote controlled helicopter with the gaze, vehicle motion, and control actions recorded. The resulting visual gaze data is classified according to the typical smooth pursuit, saccades and fixations and then incorporated into an estimation strategy.

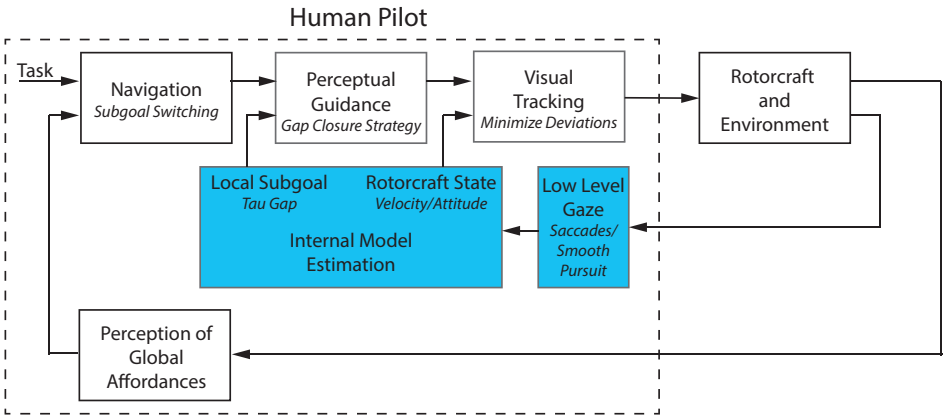


Figure 7.1: This chapter investigates gaze and how that information can be used as part of the control strategy (blue boxes).

Figure 7.2 provides an overview of the experimental setup. As discussed in Chapter 3, the lab provides capabilities for investigating human perceptual and control functions. The experiments in this work were conducted using a "third person" modality. Based on the helicopter dynamics, gaze patterns, and control inputs provided by the facility, the specific goal is to identify how the gaze is integrated with human control actions.

Moving forward, the knowledge about the coupling between perception and control modes of a human operator can be used to design more natural and intuitive control interfaces. The human operator and the automated systems provide complementary capabilities with the human able to excel at high-level reasoning, task determination, spatial mapping, and guidance while the automated systems excel at stabilization, trajectory optimization, and path following. At one end of the spectrum, consider an operator interface utilizing the gaze patterns of the pilot. This system could be integrated with a brain-computer interface such as the one in [109]. At the other end of the spectrum, the pilot could maintain full manual control with the human-machine control interface providing cues to compensate for the limitations of perception and control capabilities. For autonomous vehicles the gained knowledge could help design sensors and perceptual algorithms that operate based on extracted information.

The chapter is organized as follows. The following section provides a brief overview of pilot modeling and visual perception followed by the elements of the modeling framework adopted in the present study. Section 7.2 discusses gaze registration and classification methods. Section 7.3 describes the experiments that were used to investigate the control and guidance functions with

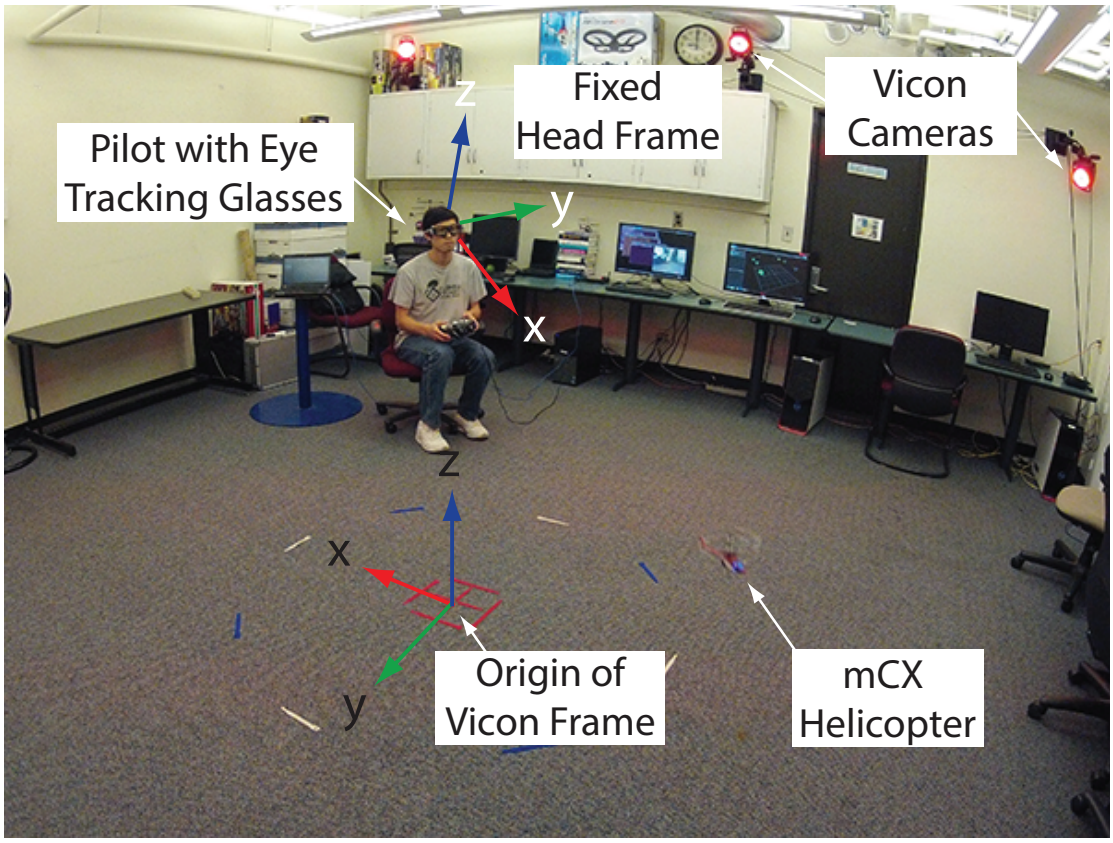


Figure 7.2: Experiment in the lab environment showing rotorcraft, Vicon motion tracking system, and gaze tracking. The problem investigated in this chapter is to identify the interactions between gaze patterns and human control actions.

a discussion of the gaze tracking behaviors that were observed. Section 7.4 presents the results of the analysis focusing on the identified visuo-motor feedback for stabilization, tracking and goal interception. Section 7.5 discusses a model for integrating gaze and control. Section 7.6 integrates the gaze information into an estimation strategy that generates output suitable for control. Finally, section 7.7 provides concluding remarks.

7.2 Gaze Processing

The eye tracking system described in Section 3.3.2 generates a gaze vector in terms of the operator's head orientation. To fully utilize the information the gaze vector needs to be registered

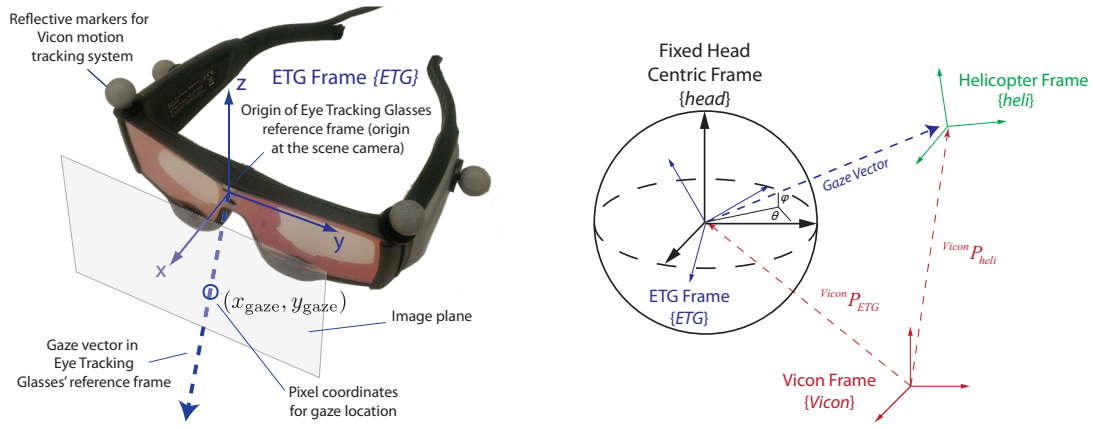
in a common reference frame and the gaze classified into it's constituent modes. This section describes the registration and classification procedures.

7.2.1 Registration of Gaze and Motion Tracking Measurements

Understanding the human perceptual processes supporting guidance and control capabilities requires linking visual gaze with the vehicle, task and environment elements. The eye tracking glasses shown in Fig. 7.3(a) provides a gaze vector relative to the pose of the pilot's head. Therefore, a registration procedure is required to put the pilot's head, helicopter, and environmental features in a common reference frame.

The determination of a reference frame for gaze is crucial for gaze classification. For instance, a 3D inertial Vicon frame is able to provide information about where human pilots are focusing, but the transition of the gaze point in this frame cannot reflect the magnitude of gaze movement. For instance, two stars are far in space but close to each other from visual perspective.

The gaze should be transformed into a reference frame appropriate for use by human decision making and motor control systems. A spherical head centric coordinate frame is proposed in [110] to describe the visual receptive field of flies and can be extended to humans. It is used in this paper and represents gaze as azimuth (θ) and elevation (ϕ) angles, as shown in Figure 7.3(b).



(a) Eye Tracking Glasses Reference Frame.

(b) Spherical Reference Frame.

Figure 7.3: Reference Frames for Generating 3D Gaze Location.

Combining the gaze tracking glasses with the Vicon motion tracking system allows for mapping 2D gaze into the 3D world and can also take tracked objects and project their location onto a spherical head centric coordinate frame as shown in Figure 7.3(b). Registration of the gaze measurements, helicopter motion, and other environmental features is achieved using the Vicon motion tracking system, which generates a common reference frame. The pose of the pilot's head is tracked by the Vicon system and used to transform the gaze vector into a fixed reference frame located at the average head position obtained during a given experimental run. The fixed reference frame uses a spherical coordinate system and maps the combined head rotation and gaze vector to azimuth (θ) and elevation (ϕ) angles. The Vicon system provides the pose of the eye tracking glasses (${}^{ETG}_{Vicon}R$, ${}^{Vicon}P_{ETG}$) and helicopter (${}^{heli}_{Vicon}R$, ${}^{Vicon}P_{heli}$) in the Vicon reference frame. Since it is assumed that there is minimal change in the position of the pilot's head during a single run only head rotational motion is considered. The θ and ϕ angles for the gaze vector in the fixed head frame can be found using:

$$\begin{aligned}\text{Image plane (IP)} &\Rightarrow \text{ETG frame (ETG)}\\ &\Rightarrow \text{Vicon frame (Vicon)}\\ &\Rightarrow \text{Fixed head centric frame (Head)}\end{aligned}$$

IP $\Rightarrow$ ETG

The reference frame defined by the ETG has its origin centered on the scene camera of the ETG as shown in Fig. 7.3(a). Gaze location is represented by the pixel location in the scene image (1280x960 resolution). Based on the field of view (FOV) of the scene camera (60° horizontal and 46° vertical), the gaze direction is converted to an angular vector in the ETG frame, $[\theta_{\text{ETG}}, \phi_{\text{ETG}}]$:

$$\theta_{\text{ETG}} = \frac{x_{\text{gaze}} - x_{\text{size}}/2}{x_{\text{size}}} \text{FOV}_{\text{hori}} \tag{7.1}$$

$$\phi_{\text{ETG}} = \frac{y_{\text{gaze}} - y_{\text{size}}/2}{y_{\text{size}}} \text{FOV}_{\text{vert}}. \tag{7.2}$$

ETG $\Rightarrow$ Vicon

Given the head pose (position ${}^{\text{Vicon}}H$ and Euler angles ${}^{\text{Vicon}}A$) captured by the Vicon system, a rotation matrix from the Vicon frame to the ETG Cartesian frame, ${}^{\text{ETG}}_{\text{Vicon}}R$, can be constructed.

The unit vector of gaze in the ETG frame, ${}^{\text{ETG}}V = [\cos(\theta_{\text{ETG}})\cos(\phi_{\text{ETG}}), \sin(\theta_{\text{ETG}})\cos(\phi_{\text{ETG}}), \sin(\phi_{\text{ETG}})]$, can be projected to the 3D Vicon frame by

$$ {}^{\text{Vicon}}V = {}^{\text{ETG}}_{\text{Vicon}}R^{T} \cdot {}^{\text{ETG}}V. \tag{7.3} $$

The depth information d cannot be captured by the ETG. It is recovered by intersecting the gaze vector with the horizontal plane of the helicopter altitude. This is formulated as

$$ d = \frac{(O - {}^{\text{Vicon}}H) \cdot n}{{}^{\text{Vicon}}V \cdot n} \tag{7.4} $$

$$ {}^{\text{Vicon}}P = {}^{\text{Vicon}}H + {}^{\text{Vicon}}V \cdot d \tag{7.5} $$

where O is the helicopter position in the Vicon frame and n is the vertical vector.

Vicon $\Rightarrow$ Head

The fixed head frame is determined as the average head pose during an experimental run. Gaze registration is accomplished through the mapping from the Vicon frame to the fixed head frame:

$$ {}^{\text{Head}}P = {}^{\text{Head}}_{\text{Vicon}}R \cdot ({}^{\text{Vicon}}P - {}^{\text{Head}}H), \tag{7.6} $$

representing gaze as

$$ \theta_{\text{Head}} = \tan^{-1}(x_{\text{Head}}/y_{\text{Head}}) \tag{7.7} $$

$$ \phi_{\text{Head}} = \sin^{-1}(z_{\text{Head}}/d). \tag{7.8} $$

With the fixed head frame serving as the natural frame for humans' perceptive and cognitive processes, the signal of visual stimuli (including the helicopter, the task configuration and the environment cues) can also be transformed to this frame with (7.6), (7.7) and (7.8). This enables the study of the relationship between gaze focus and visual stimuli, and provides additional features for gaze classification, which is discussed in the following section.

Project Helicopter to Fixed Frame

The spatial position of visual stimulus (e.g., helicopter, start and target locations, etc.) can be projected to the same coordinate system. Assuming unit radius, the gaze vector can be converted to 3D Cartesian coordinates, ${}^{\text{mhead}}P$ in a moving head frame. The rotation matrix

from the Vicon frame to the moving-head Cartesian frame, ${}^{\text{mhead}}_{\text{Vicon}}R$, can be constructed with the current head rotations (yaw, pitch and roll). Similarly, the rotation matrix from the Vicon frame to the fixed-head Cartesian frame, ${}^{\text{fhead}}_{\text{Vicon}}R$, can be constructed with the averaged head pose. The unit gaze vector in the head-fixed Cartesian frame can then be written as:

$$ {}^{\text{fhead}}P = {}^{\text{fhead}}_{\text{Vicon}}R \cdot {}^{\text{mhead}}_{\text{Vicon}}R^{T} \tag{7.9} $$

It allows for mapping 2D gaze into the 3D world and can also take tracked objects and project their location onto a spherical head centric coordinate frame.

7.2.2 Gaze Classification

Gaze is the coordinated motion of eye and body (mainly head). Fixating a stationary object while turning the head can have similar eye movement as pursuing a moving object while holding the head still. Therefore, the measurement of both head and eye motion is required to classify gaze.

Basic eye movement is comprised of three components: saccades, fixations and smooth pursuits, each having distinct kinematic characteristics. Saccades are the fast eye movements of small durations used to redirect the eye to a new location [111]. Fixations take place when the gaze is stabilized on typically stationary points [83]. Their duration spans longer time intervals. Smooth pursuits are eye movements when the gaze follows moving visual stimuli [112].

These three basic eye movements can be classified according to their kinematic characteristics, more specifically, by setting the respective thresholds of velocity and time duration. For instance, the lower saccade speed limit for amplitudes of 5°, 10°, 20°, and 30° were determined to be 145°, 196°, 213°, and 227° per second. These characteristics were obtained by analyzing factors such as abduction, centering, eccentric, and across-the-center refixations [113].

During fixations and smooth pursuits, micro-saccades are employed as a correction mechanism to stabilize the retina and prevent motion blur, so that high-quality visual information can be acquired [83, 114]. During saccades, the visual system is essentially blind due to the motion blur.

7.3 Experiments

Experiments were conducted to investigate the pilots' control capabilities. For stabilization, a hover task was used; for tracking, a circle tracking task; and for goal interception, a step task. Descriptions of the hover task and circle task can be found in Section 5.2.1 and Sections 5.2.3 respectively. Fig. 7.4 shows an illustration of the step task used that is a variation of the one given in Section 5.2.2. Each test flight the pilot was sitting about $2m$ behind the center of the task space and was required to remain stationary during the task. Each task was performed by 4 student test pilots with skill levels varying from moderate to highly skilled. Before each session, the pilot was allowed to practice briefly before beginning a trial. The analysis in the following sections shows results for one pilot.

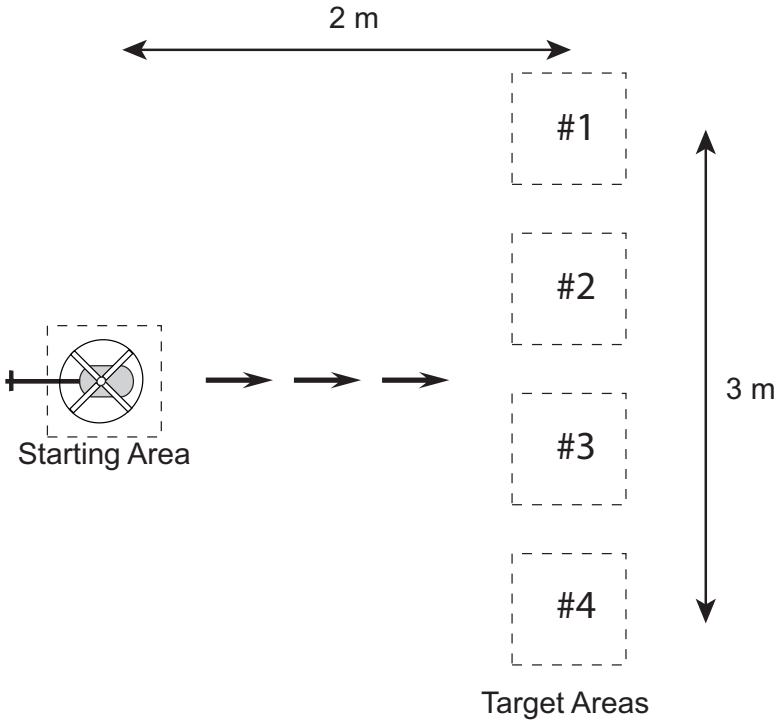


Figure 7.4: Experimental step task used to investigate the low-level control and guidance functions.

7.3.1 Stabilization – Hover Task

In the hover task (Fig. 5.4), the subjects were instructed to maintain the helicopter within a circle marked on the floor for 60 seconds. In addition, the objective was to achieve the most steady hover, i.e., to minimize the velocity fluctuations. Further, they were asked to keep the helicopter facing away from their body.

Figure 7.5 shows the gaze decomposed into a density plot of the smooth pursuit points and step changes for saccades for a small hover area ($0.25m$ in diameter). The gaze consisted of primarily smooth pursuit points with a small number saccades to the center of the constraint area (marked on the ground).

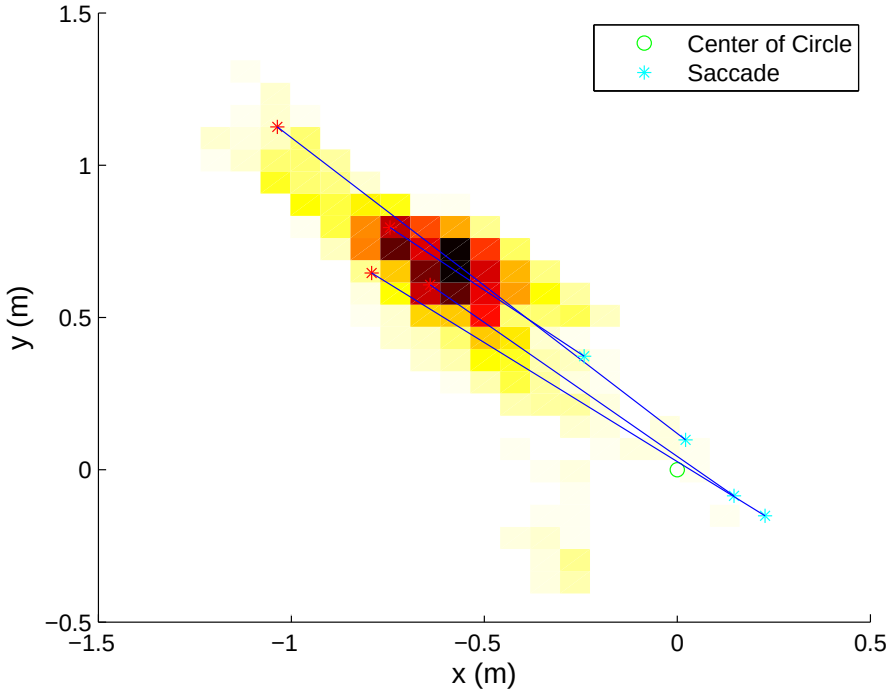


Figure 7.5: Gaze decomposition for the hover task within a small boundary area. The gaze is shown in the task space with smooth pursuit points shown as a density plot and saccades as gaze steps.

7.3.2 Interception – Step Task

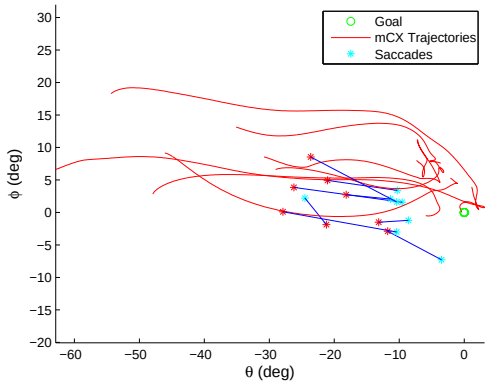
The step task (Fig. 7.4) allows isolating the processes of the perception-action loop along a single dimension. The pilots were instructed to start from a stable hover over the starting area and then perform an acceleration/deceleration maneuver ending in a stable hover over one of four target areas. The latter was specified randomly at the initiation of each trial. Random goal specification was implemented to reduce effects of accommodation.

Multiple experimental trials were conducted. Fig. 7.6(b) shows the gaze and helicopter velocities for one trial. During most of the experimental flight time the pilot's gaze operated in pursuit mode and tracked the helicopter closely. The gaze velocity during pursuit follows the velocity profile of the helicopter trajectory and suggests that the gaze can provide the measurement of the velocity used for feedback control.

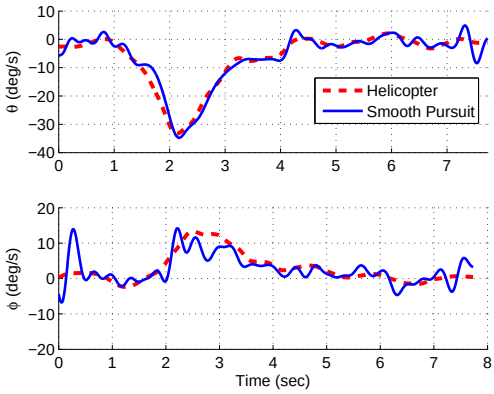
In addition, saccades take place systematically as the helicopter approaches the goal. Fig. 7.6(a) shows the saccades from multiple step trials. The data has been transformed into a reference frame with the goal at the origin and the starting locations along the negative θ (azimuth) axis. In the transformed frame the rotorcraft trajectory moves from left to right during a trial. The saccades provide a measure of the distance remaining to the goal location. This is consistent with what is expected from tau theory. Specifically, the saccades are measuring the gap that is being closed for the task.

7.3.3 Tracking – Circle Tracking Task

In the circle tracking task (Fig. 5.4), the pilots have to track a circle (radius $75cm$) marked on the floor while minimizing deviations and holding the helicopter heading constant. Fig. 7.7 shows a single isolated run with the gaze decomposition into smooth pursuits and saccades. For the majority of flight time the gaze operated in pursuit mode and matched the helicopter trajectory. Saccades occurred periodically with the saccade moving the gaze to a future point on the circular trajectory. The significance and purpose of saccades are less obvious in the tracking task than in the goal interception task.



(a) Trajectories and saccades for goal interception.



(b) Smooth pursuit.

Figure 7.6: Step task results: (a) Saccades during step experiments (multiple trials), and (b) gaze pursuit mode and rotorcraft velocities in head frame.

7.4 Models

The experimental results provide evidence that during remote operation, the gaze, helicopter dynamics, and control are tightly coupled. The next step is to determine models that describe the gaze dynamics' role as part of the helicopter control mechanisms.

7.4.1 Gaze Modalities Summary

The decomposition of the gaze trajectory into smooth pursuits and saccades reveals two primary patterns. First, for stabilization and tracking, the pilot needs information about the helicopter

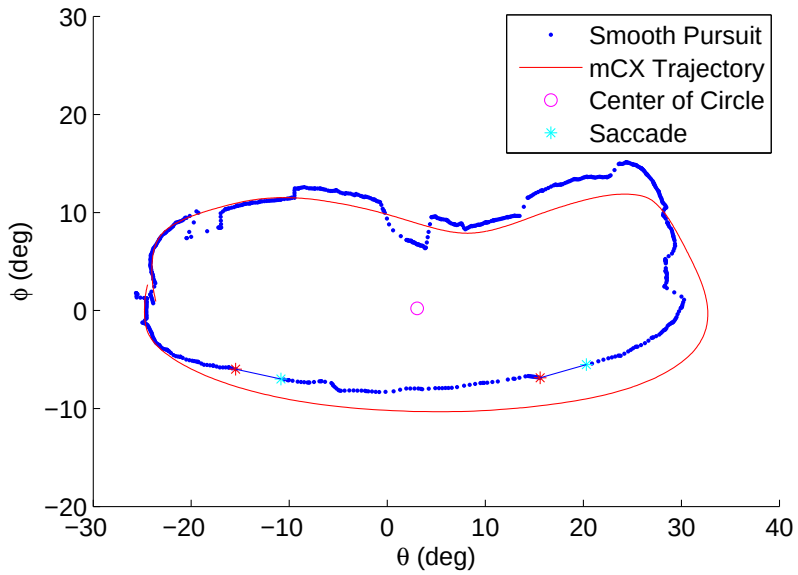


Figure 7.7: Circle task saccades and pursuit for a one of the circle trajectory.

pose and velocity. Smooth pursuit gaze trajectories follow the helicopter trajectories as shown in Fig. 7.7. The gaze velocities during smooth pursuit match closely with the helicopter velocities (see 7.6(b)). This information can be used for closing the velocity loop in Fig. 7.1. Second, guidance control is clearly mediated through saccades. Fig. 7.5 and Fig. 7.6(a) show that the saccades move between the gaze pursuit trajectory and the target location. The saccades measure the gap (from tau theory) that needs to be closed or maintained for the task thus providing feedback at the guidance control level.

In general tasks, the information necessary to close the guidance and control loops would be available from the scene's visual content, the peripheral vision and the information used for the active control of gaze. During typical operation the lower-level control modes require precise, high-bandwidth information, therefore the assumption is that the peripheral vision plays a secondary role. This is supported by the high correlation between helicopter control behavior and the gaze pattern. For guidance tasks in unstructured environments, we would expect that peripheral vision plays a more significant role since more global information about the environment and task elements must be acquired. This aspect is currently being studied and is beyond

the scope of this chapter.

7.4.2 Pursuit Model

The low-velocity, smooth pursuit mode is primarily operating from visible visual cues within the high-resolution region. To analyze the interaction between gaze and control during smooth pursuit, the transfer function between the gaze and helicopter velocity was identified from frequency responses extracted from input and output data. Fig. 7.8 shows the identified transfer function. The input x, which is the stimulus for pursuit is the helicopter velocity, and the output y is the gaze velocity. Frequency response and coherence estimates are computed from

$$T_{xy}(f) = \frac{P_{yx}}{P_{xx}}; \qquad \gamma_{xy} = \frac{|P_{yx}^2|}{|P_{xx}P_{yy}|}$$

where P_{yx} is the cross- and P_{xx} the auto-power spectral densities. At low frequencies (< 1Hz), the gain is one and the phase is close to zero, indicating that gaze provides a near perfect velocity measurement. The associated coherence γ_{xy} remains large and confirms a linear input-output relationship.

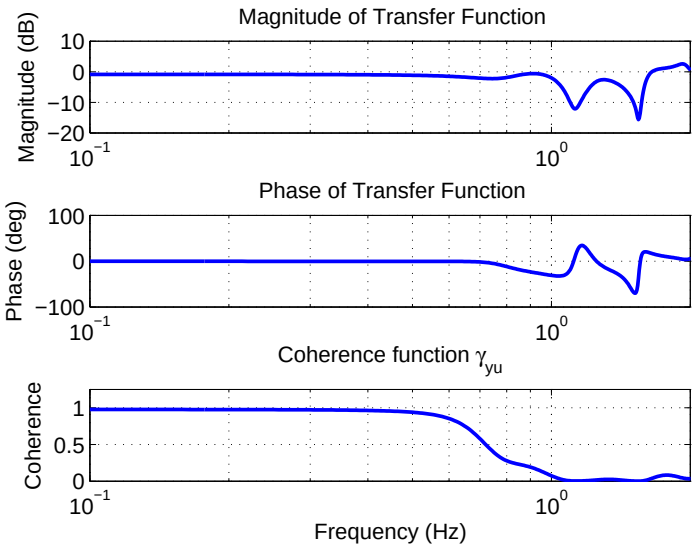


Figure 7.8: Identified transfer function between helicopter and gaze motion.

7.4.3 Saccade Model

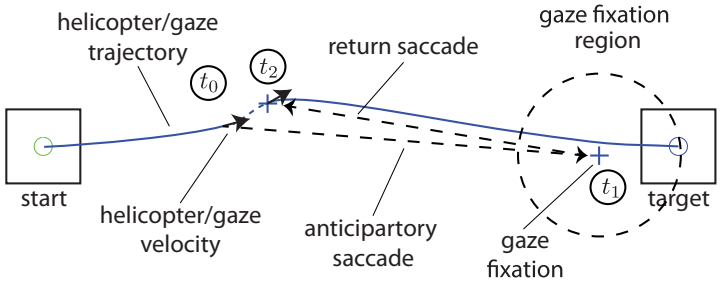
High-velocity saccade motion measures distances to features in the visual environment beyond the currently visible area. Saccades, therefore, provide measurements that are needed to guide motion including the tau gaps. The motion gap can then be used to generate a velocity reference.

The saccade mode generates information about the tau gap. The key variables for generating the tau gap information are shown in Fig. 7.9(a). The figure shows the three-stage sequence starting with smooth pursuit followed by the saccade to a fixation point (t_1) near the desired goal location and concluding with another saccade returning to the helicopter (t_2). This sequence may be initiated multiple times. For a trained pilot moving to a previously visited target, saccades may not be utilized. Fig. 7.9(b) shows the position distributions during the step task in Fig. 7.6(a) based on mean and standard deviation for the three times $t_0 - t_2$. It is interesting to note that the saccades stop short of the goal by about 10 deg, which is enough to bring the goal within range of the central eye field. Table 7.1 provides statistics for the key variables in the gaze interaction model.

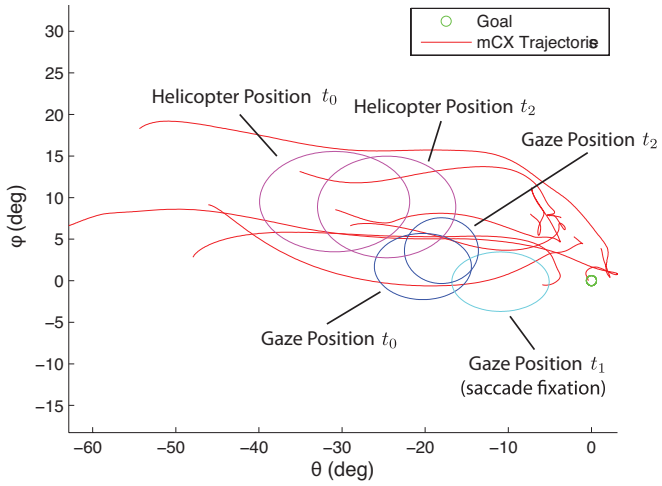
Table 7.1: Mean and Standard Deviation for Key Variables in Gaze Interaction Model for the Step Task.

	Mean		Stdev		
Variable	θ	ϕ	θ	ϕ	unit
Gaze Position 0	-20.28	1.71	5.84	3.98	deg
Heli Position 0	-30.89	9.51	9.03	6.04	deg
Gaze Velocity 0	11.18	-2.53	21.15	2.41	deg/s
Heli Velocity 0	21.44	-3.25	8.04	3.89	deg/s
Gaze Position 1	-10.94	-0.12	5.85	3.58	deg
Gaze Position 2	-18.05	3.62	4.45	3.95	deg
Heli Position 2	-24.63	8.87	8.31	6.11	deg
Gaze Velocity 2	12.16	-5.36	17.36	6.69	deg/s
Heli Velocity 2	20.71	-0.96	7.21	3.49	deg/s

Two factors are critical for determining when a saccade is triggered. The first is the size of the visual angle between the current gaze focus of attention and a task element. When the high visual acuity area of the gaze is close to a task element visual information is captured about the elements position. As the gaze focus moves farther away information about the task element becomes uncertain and eventually triggers a saccade. The question to be answered is how close does the gaze focus need to be to capture information on the task element. Based on the anatomy of the human eye, the fovea has by far the highest visual acuity and accounts for 5 degrees of



(a) Key variables involved in saccades to a target.



(b) Statistics extracted from step task.

Figure 7.9: Description of the saccade dynamics during the step task. The ellipsoids are based on mean values for the position variables along with the 1σ error bounds for the step task.

the visual field, the parafovea around 8 degrees, and the perifovea 18 degrees. Fig. 7.10 shows a distribution of the saccades generated during a hover task with different visual angles between the vehicle and target on the ground. The visual angles are achieved by hovering at different heights. The figure shows that for 10 degrees (the fovea and parafovea regions) or less few saccades are generated indicating sufficient information on the target is available and saccades are infrequent. For gaze angles from 10-20 degrees (the perifovea region) the frequency of saccades increases. Beyond 20 degrees the human pilot is continually generating saccades to measure the target position relative to the vehicle.

The second factor in triggering a saccade is the time since a task element was last observed.

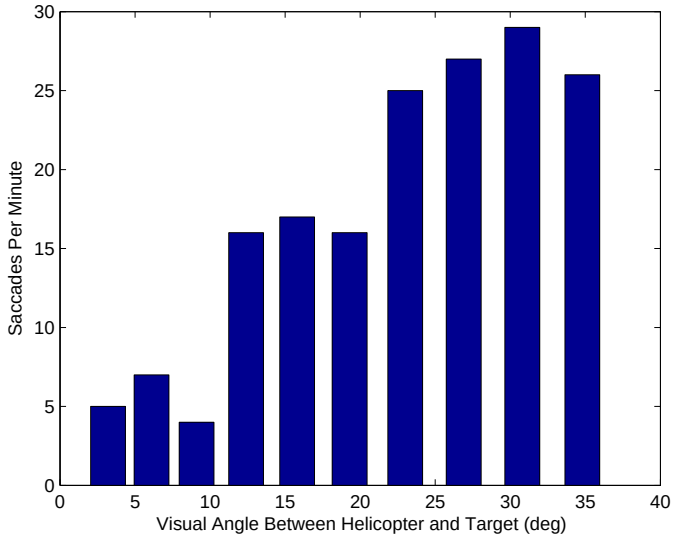


Figure 7.10: The saccade frequency during hover experiments for different heights (visual angle between the helicopter and target on the ground).

According to work done by Brown [115] and numerous others since, working memory decays after around 15 seconds. The retention interval decreases as more items of information need to be remembered [97]. To test how long a pilot can keep track of a distant target location a step experiment was conducted with the pilot hovering over the starting location for differing lengths of time before initiating the maneuver. As the hovering time increases the chance for a saccade to be triggered increases. Fig. 7.11 shows the saccade frequency for increasing time durations since the target was last observed. In the figure the number of saccades per trial increases to nearly 1 or more indicating that enough uncertainty has accumulated since the pilot last observed the target that a saccade is triggered.

7.5 Integrated Gaze and Control Model

The block diagram in Fig. 7.12 describes a notional model of the primary gaze and control functions and their integration based on the teleoperation experiments. To visually track an object such as a helicopter, coordinated eye and head movements must be generated (‘Head Eye System’). In smooth pursuit, the gaze keeps the visual stimuli guiding the pursuit near the

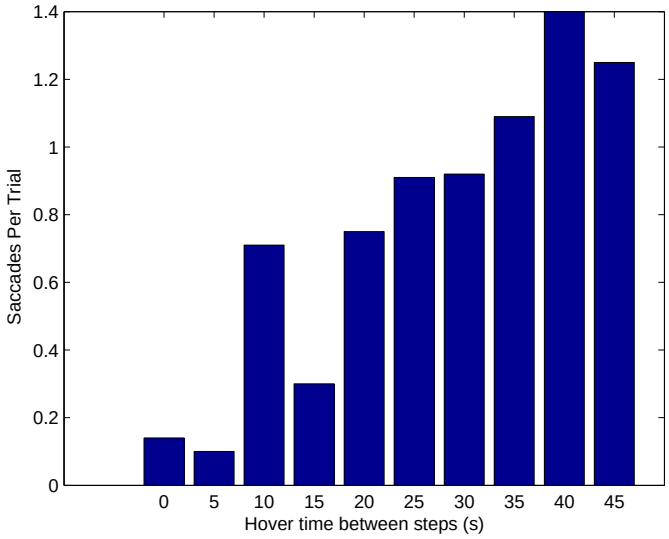


Figure 7.11: The saccade frequency during step experiments for different lengths of hover time over the starting position (gives increasing lengths of time since the target location was last observed as part of the task execution).

center of the eye field where the eye's resolution is highest (about 10 deg region). For goal interception the saccades provide anticipatory information about the goal location.

During visual tracking and guidance, pose and velocity measurements needed to control the helicopter are derived from the motor control signal driving the head/eye system. This information is first integrated within an 'Internal Model' that simultaneously estimates the goal and rotorcraft state (position and velocity). This information is then used to generate a control action (via the 'Motor Control' system) utilizing both open-loop and closed loop strategies. The control output can be mapped into the helicopter frame using different strategies to overcome the lack of depth information, for example assuming a constant height. For the goal interception, the tau gap extracted from the saccade information provides anticipatory information. Under conditions, entire segments of the trajectory can be generated and implemented in open loop. Finally, as highlighted in the block diagram, the different components operate within different reference systems. In some cases, such as in the 'Internal Model' and 'Motor Control', two coordinate systems are most likely used in parallel.

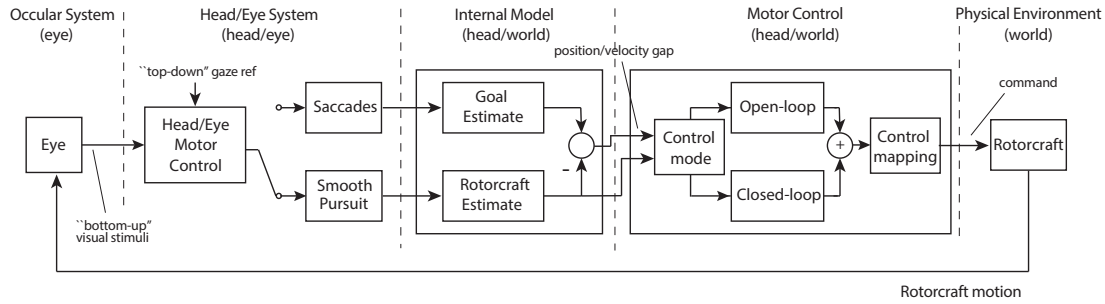
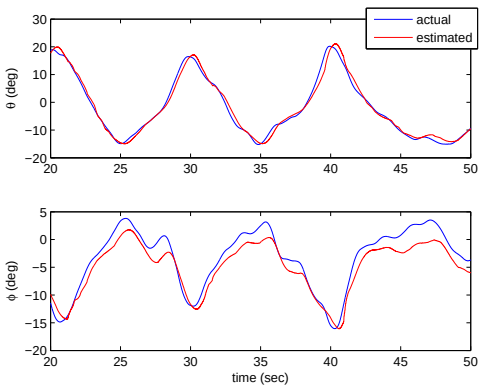


Figure 7.12: Block diagram of a notional model of the gaze control and helicopter control integration.

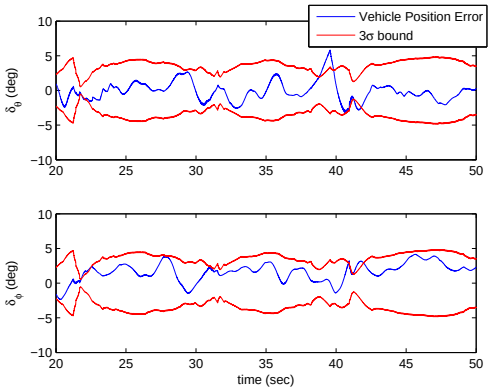
7.6 Estimation of Vehicle and Goal States

The 'Internal Model' in Fig. 7.12 can be described using standard state estimation techniques. The key information that needs to be estimated is the vehicle state and the position of the current task elements. A simple approach for estimating these values is achieved using a constant velocity Kalman filter designed to estimate the vehicle position and velocity. Additional states are added to the estimator to track the position of the target locations. Measurement updates are generated by the visual system. During smooth pursuit, the gaze location provides adequate information for tracking the vehicle as seen in Fig. 7.13(a). To estimate the goal position, the visual system identifies the θ and ϕ angles of the target location in the head reference frame. The distance of the visual features from the center of the fovea determines the measurement covariance. This results in high accuracy measurements when the gaze is focused on an object and low accuracy at approximately 15 *deg* away from the center of vision. The estimation errors along with 3σ bounds are shown in Fig. 7.13(b) and Fig. 7.13(c) for the vehicle position and one of the goal locations. As discussed earlier, information about environmental features grows more uncertain the longer the time since the feature has been observed. To capture this characteristic, a forgetting factor λ is incorporated into the estimator and causes the error bounds for the goal position to gradually increase during the transition between locations unless a saccade to the goal occurs. Once the vehicle is near the goal position, the error bounds decrease since the goal is close enough to the high acuity visual area to provide update information.

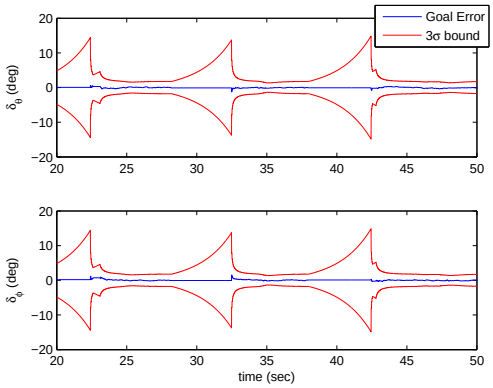
The EKF time update equations, for a task of flying between two positions marked on the floor, are of the form given in (7.10). The states x are the azimuth (θ) and elevation (ϕ) angles for the helicopter and goal. The Jacobian calculations for the state transition matrix and the



(a) Vehicle position.



(b) Vehicle position error.



(c) Goal position error.

Figure 7.13: Results for the vehicle and goal position estimation using the gaze data as measurement.

matrix characterizing model uncertainty are given by Φ and Q. The update equations for the

state and covariance estimate P are denoted by the last two equations.

$$
\begin{aligned}
x &= [\theta_{heli}, \dot{\theta}_{heli}, \phi_{heli}, \dot{\phi}_{heli}, \theta_{goal}, \phi_{goal}] \\
\Phi &= \begin{bmatrix} 1 & dt & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & dt & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \\
Q &= \begin{bmatrix} dt^3/3 & dt^2/2 & 0 & 0 & 0 & 0 \\ dt^2/2 & dt & 0 & 0 & 0 & 0 \\ 0 & 0 & dt^3/3 & dt^2/2 & 0 & 0 \\ 0 & 0 & dt^2/2 & dt^2/2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.01 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.01 \end{bmatrix} \\
\hat{x}_{k+1|k} &= \Phi \hat{x}_{k|k} \\
P_{k+1|k} &= \lambda \Phi P_{k|k} \Phi^T + Q
\end{aligned}
\tag{7.10}
$$

The EKF measurement update equations are given in (7.11). The observation vector $\hat{z}$, observation covariance matrix H, measurement covariance matrix R, Kalman gain K, and updates for the state vector and state covariance matrix are given by:

$$
\begin{aligned}
\hat{z} &= H \hat{x}_{k+1|k} \\
H &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \\
R_k &= \begin{bmatrix} \sigma^2_{meas}(k) & 0 \\ 0 & \sigma^2_{meas}(k) \end{bmatrix} \\
K &= P_{k+1|k} H^T (H P_{k+1|k} H^T + R_k)^{-1} \\
\hat{x}_{k+1|k+1} &= \hat{x}_{k+1|k} + K(z_m - \hat{z}) \\
P_{k+1|k+1} &= (I - KH) P_{k+1|k} (I - KH)^T + K R_k K^T .
\end{aligned}
\tag{7.11}
$$

Other factors such the number of tracked objects near the center of the fovea as well as the size of the objects could also affect the accuracy of the measurement, but are not considered in this chapter.

7.7 Conclusions

The goal of this chapter was to characterize the coupling between gaze and human control when using a multi-loop hierarchical control architecture. Experiments for simple tasks that exercise the basic control capabilities were conducted. The experimental results demonstrate how the gaze patterns can be decomposed into smooth pursuit and saccades. These visual cues were then analyzed in conjunction with the helicopter dynamics and pilot inputs to determine the primary visuo-motor feedback mechanisms in the multi-loop human control system. The smooth

pursuit gaze behavior was shown to provide a measure of the rotorcraft velocity needed for its stabilization and regulation. The saccades, on the other hand, provide anticipatory information for guidance both in the interception of a precise goal point and precise trajectory tracking. In the step task, the saccades measure the gap to the goal location, which is consistent with tau theory. In the circle task, the saccades provide predictive information about the future trajectory as expected from the control model for precise trajectory tracking. These results provide the key elements for the human visuo-motor control law used by operators for precise teleoperation. Using the information extracted from gaze as updates an estimation model was designed to generate the signals needed for control.

Chapter 8

Application Demonstration

8.1 Introduction

A significant aspect of the human visual experience is due to the head-eye coordination. At the same time, the gaze control mechanisms are closely involved in the guidance and control of movement. The video display in current teleoperation setups does not account for the natural head-eye interactions and therefore can adversely impact performance. This chapter investigates automating remote camera positioning based on the operator gaze behavior. The camera is mounted on an actuated gimbal that uses real-time gaze measurements to mimic human head movement. A second application example implements control augmentations to demonstrate how the gaze can be used as part of the vehicle control architecture. Fig. 8.1 shows the components relevant to each of the two application examples.

The primary contribution of this chapter is the development of a teleoperation system that integrates eye tracking as part of the control architecture. The example applications implement models of gaze to provide control of an actuated camera and control augmentations to simplify the tasks. Secondary contributions include: i) the method generate saccades from the eye tracking data on the teleoperation display, and ii) implementation of the head-eye motion models and control augmentations in a real-time teleoperation system.

The remainder of the paper is organized as follows. Section 8.2 gives and overview of the intended application and overall system description, followed by a brief review of related work on teleoperation systems. Section 8.3 describes the gaze mediated camera control system and

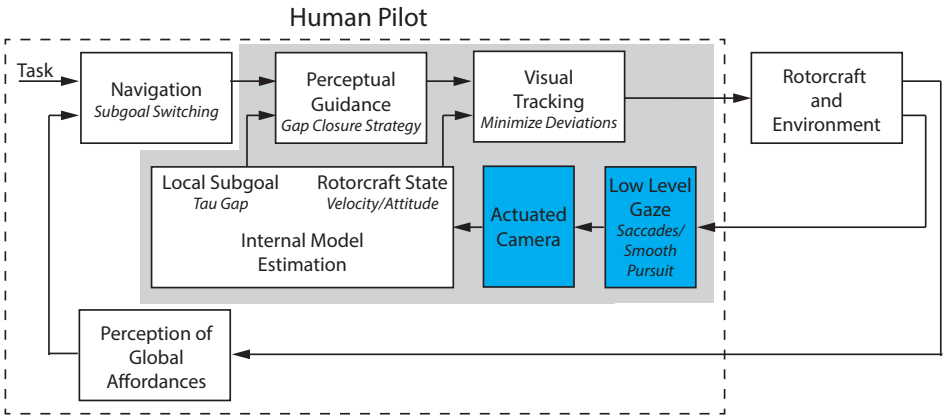


Figure 8.1: The components implemented for each example application are highlighted. The blue boxes are the components for the actuated camera and the blocks inside the gray boundary are implemented in the augmented control example.

presents results from the experimental validation including the comparison between teleoperation configurations using a task difficulty rating metric. Section 8.4 describes an example application with control augmentations along with experimental results. The paper's conclusions are given in Section 8.5.

8.2 Background and Overview

Teleoperation systems and the issues related to their successful implementation have been studied extensively. Autonomous or semi-autonomous operations have made important progress, but for the foreseeable future human teleoperation will remain a critical modality in particular for interactive tasks such as surgery or vehicle operation in complex environments.

8.2.1 Application Overview

The present work focuses on teleoperation systems for remote-control tasks. Experiments are conducted using a miniature rotorcraft as shown in Fig. 1.3(b). The teleoperation camera can be rotated to change the view of the environment or track the miniature helicopter during flight. The system uses a GoPro camera mounted on a tripod with a 3-axis motorized gimbal that provides controlled rotation of the camera to mimic head movements. A live video feed from the camera is shown on the teleoperation display. The pilot sits in front of the teleoperation

display (facing away from the lab environment), and operates the rotorcraft using only the visual information from the display. The pilot's gaze location on the display is captured using a gaze tracking device and is used in conjunction with models of the head-eye system to automate control of the camera orientation. Note that the helicopter mainly serves as a platform for investigation. The general approach could be used in other applications where visual guided motion is important, such as robotics, video games, or telesurgery.

8.2.2 Related Work

Approaches for teleoperation have been proposed to help overcome problems resulting from a limited field of view (FOV). Voshell [116] developed a multi-camera system that provided the operator a wrap around effect in order to increase the FOV. In [117, 118] FOV issues were investigated with the conclusion that performance could be optimized by providing the added capability to change the FOV. Zhu [119] actuated a camera based on gaze using a simple "move to the center" method that actuates the camera to keep the gaze in the middle of the image. An approach using gaze as a control input overlaid boxes on the display that allowed the user to select actions by focusing the gaze on different areas [120]. Also relevant are applications of human head-eye models for control of a robotic head [121].

8.2.3 Approach Overview

To this day, limited research has been devoted to the investigation of the use of gaze as an integral part of the control strategy in teleoperation [12]. In [12], the coupling between gaze modes and human control actions was investigated. Experiments were conducted using a miniature helicopter while data about the vehicle state, control actions, and operator gaze were recorded. The relevant control information was extracted by decomposing the gaze into saccades and smooth pursuit and examining the gaze patterns. In the following, the understanding about the gaze interaction is used to optimize the teleoperation system. First, to optimize the live video display and second to optimize the control modality. The goal is to provide natural experience and exploit the head-eye control mechanisms. The effectiveness of the approach is evaluated comparing the operator workload for different teleoperation configurations.

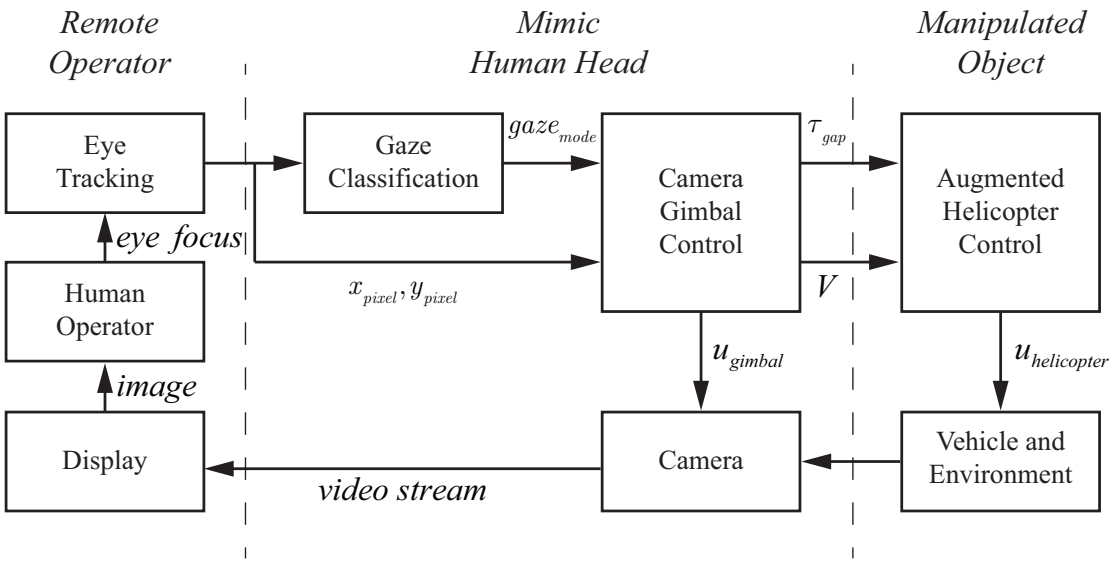


Figure 8.2: Components of the gimbal control architecture.

8.3 Gaze Mediated Camera Control

To mimic head movements the approach shown in Fig. 8.2 was utilized. As seen in the figure, the systems for gaze classification and gimbal control need to be defined. In addition, the gimbal control model needs to account for the different modes of gaze operation, i.e. saccades and smooth pursuit. This section describes experimental results and modeling of the components highlighted in blue in Fig. 8.1.

8.3.1 System Overview

The block diagram of the gimbal control architecture is shown in Fig. 8.2. The key components are the eye tracking device, the gaze classification algorithm, and the gimbal control system. The eye tracking device extracts the location of the operator's focus of attention on the teleoperation display with the gaze classification algorithm from Section 7.2.2 determining whether the gaze is currently in a fixation, saccade, or smooth pursuit mode. The gimbal control system takes the gaze location and mode to generate control signals that manipulate the camera orientation to conform to natural head movements. In addition, the gimbal control system also generates information for the control of the rotorcraft in the form of the rotorcraft velocity (V,

the estimated value of the human operator efference copy of helicopter velocity) and the gap to close (τ_{gap}) with respect to the current subgoal for the task.

8.3.2 Gimbal Control

Gimbal control that mimics the human head-eye system requires different control approaches for the low speed visual tracking of smooth pursuit and the high speed gaze motion of a saccade. The appropriate gimbal control model is activated based on the current gaze mode (saccade or smooth pursuit). This section describes the gimbal control architectures implemented for generating saccades and performing smooth pursuit that emulates the human head. The experiments to identify the head-eye characteristics were conducted with the operator directly observing a moving laser pointer while wearing eye tracking glasses.

Control for Saccades and Fixation

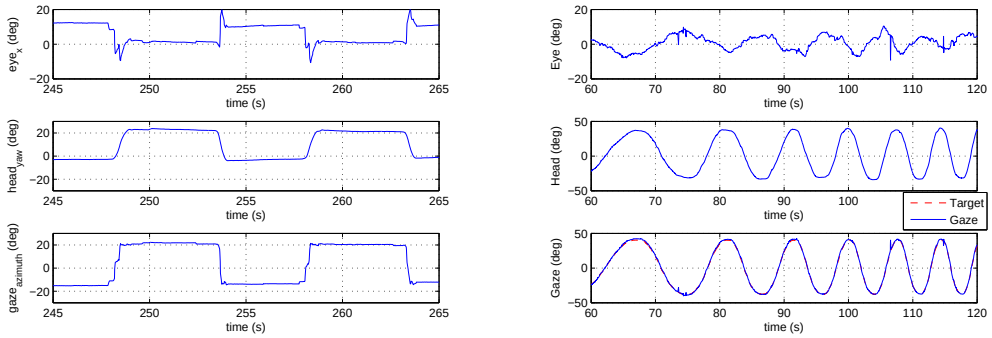
To model the human head-eye system response to gaze shifts of varying size, an experiment was conducted using the laser pointing system described in section 3.3.5. The laser point on the ground was controlled to produce steps of varying size that the human subject was instructed to follow with their eyes. An example of the resulting eye, head, and gaze patterns is given in Fig. 8.3(a). As seen in the figure, the eye has an initial fast response to a gaze shift while the head responds slower. The combined head-eye motion generates a clean and repeatable gaze shift.

The model used to control the gimbal during gaze shifts is shown in Fig. 8.4. The model is based on Fig. 2.8 and generates a fast head velocity that is determined by the size of the gaze shift. In addition, the controller captures the slow phase response necessary to correct the head position at the end of the gaze shift. The control of each dimension (azimuth and elevation) is treated independently with Fig. 8.4 showing one dimension.

The size of the gaze shift (S) is identified from the amplitude of the initial eye saccade. In [5] it was shown that the amplitude of the eye saccade and the size of the gaze shift are linearly related. Based on experimental data a linear model was identified to convert the eye amplitude to a gaze step size (G) as follows

$$G = 1.8S - 1.17. \tag{8.1}$$

Fig. 8.5 shows the results from experimental trials that captured the initial amplitude of the eye



(a) Head-eye-gaze time history for the saccade experiment using eye tracking glasses to follow a moving laser point (in the body frame).

(b) Head-eye-gaze time history for the smooth pursuit experiment using eye tracking glasses to follow a moving laser point.

Figure 8.3: Experiments for saccades and smooth pursuit gaze tracking.

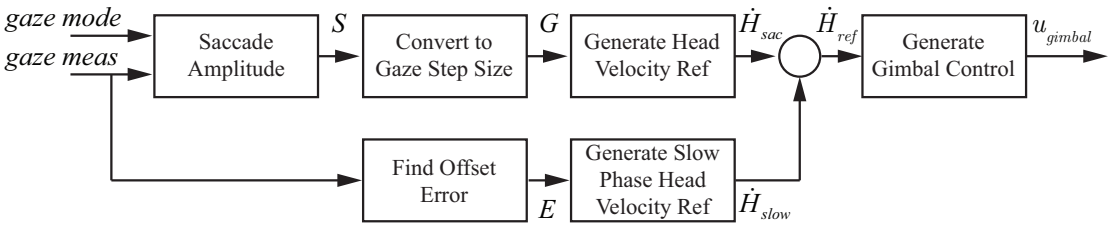


Figure 8.4: Block diagram showing the configuration of the saccade and fixation gimbal controller.

saccade plotted against the size of the resulting gaze shift. The linear fit shown in the figure had an R^2 value of 0.81. The fit shown in the figure converts the eye amplitude to a gaze step size. To generate the head velocity reference the gaze step size is multiplied by a constant $\dot{H}_{sac} = 1.25G$. The slow phase head velocity reference in Fig. 2.8 uses a constant gain to convert the distance of the eye from the center of the display into a corrective head velocity. The slow-phase head velocity reference is attenuated for large eye offsets to disable the correction during large gaze shifts.

Control for Smooth Pursuit

Experimental evaluation of the head-eye system operation in the smooth pursuit mode was conducted using the laser pointing system. In the experiment a chirp signal was applied to

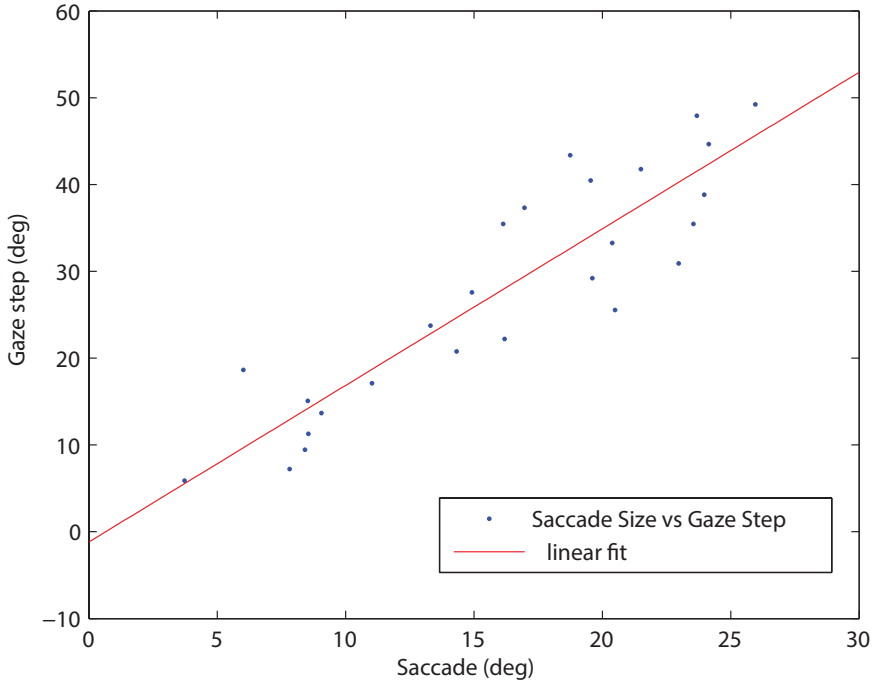
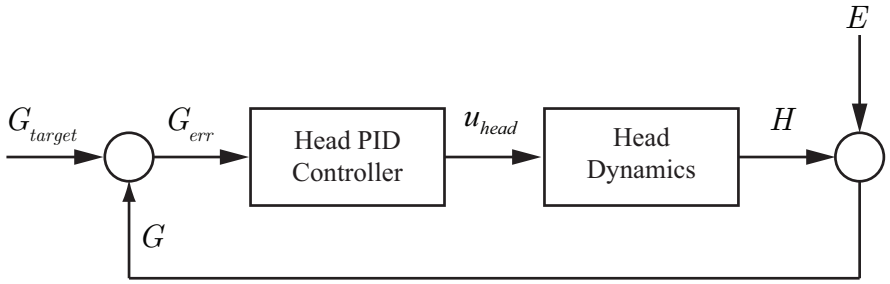


Figure 8.5: Model of initial eye saccade size and size of gaze step.

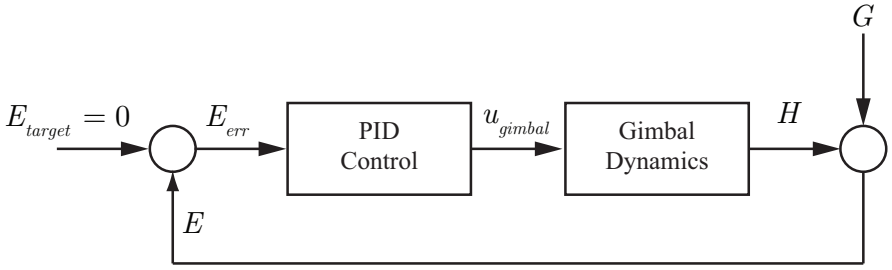
move the laser point at increasing velocity along a trajectory that caused yaw movement of the head. An example of the resulting eye, head, and gaze patterns is given in Fig. 8.3(b). The figure shows that the eye and head coordinate to track the moving point with the head providing the majority of the motion.

The smooth pursuit mode of head-eye coordination follows the model shown in Fig. 2.9. The head control and dynamics for this type of system is shown in Fig. 8.6(a). In the system, the gaze error (G_{err}) is used in closed-loop control to drive the head along a trajectory that tracks the desired target. The combined head-eye motion provides the gaze.

From the experimental data collected during the experiment shown in Fig. 8.3(b), the frequency response was calculated for the open-loop from G_{err} to H using the FRESPID tool in CIFER [108]. A transfer function was fit to the frequency response using the NAVFIT tool with the result shown in Fig. 8.7. The transfer function assumes a PID form ($K_p + K_d s + K_i s^{-1}$) for the control and second order dynamics for the head. The objective for the gimbal control system is to match the dynamic response of the human head-eye system in order to provide a response



(a) Block diagram showing the structure of the head-eye control during smooth pursuit.



(b) Block diagram showing the configuration of the smooth pursuit gimbal controller.

Figure 8.6: Block diagrams for smooth pursuit control.

that feels natural to the operator.

To implement gimbal control for the smooth pursuit mode the block diagram from Fig. 8.6(a) is modified to the form in Fig. 8.6(b). The assumptions used to make the changes are that the eye error (E_{err}) is equivalent to the gaze error (G_{err}), which holds true while the target is in the field of view and that the head control system functions to keep the eye centered on the display. A PID controller was implemented rather than a more advanced control strategy in order to match the existing model of human smooth pursuit presented in Section 2.6.1.

8.3.3 Experimental Results

This section presents the results from experiments conducted using the gaze mediated teleoperation system.

Environment Sensing with Saccades

When a saccade is detected, a fast gimbal motion is executed to reposition the camera to point at the area of interest. An experiment was conducted with a single starting location and multiple

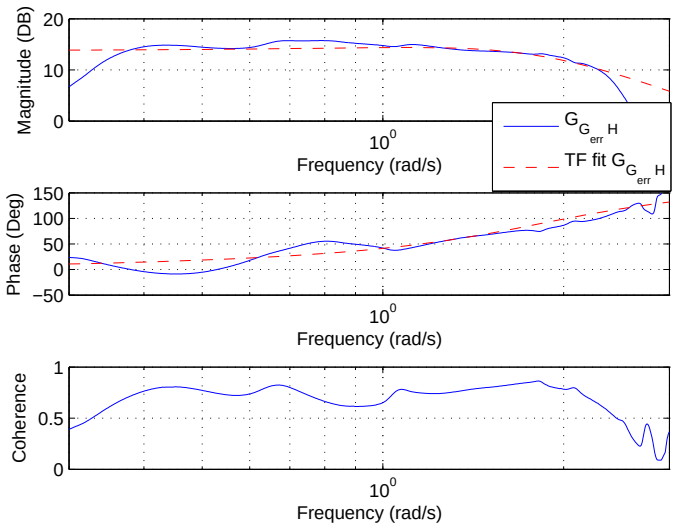


Figure 8.7: Frequency response and identified transfer function fit for open-loop dynamics from G_{err} to H.

possible goal locations (for details see [12]). While hovering over the start position, the operator was instructed to fly to one of four possible goal locations. Due to the operator's uncertainty about the goal locations, saccades were generated to quickly verify the positioning of the goal before beginning the maneuver. Fig. 8.8 shows an example of the head-eye-gaze during the experiment. The figure shows three trials of the pilot flying the vehicle from a start location to one of the possible targets and then returning to the original position. At the start of each trial, a saccade is generated when the pilot is informed of the next target.

Performance Evaluation

This section compares the proposed teleoperation system with alternative configurations. The first configuration used a camera with a fixed field of view while the second option had an additional operator manually controlling the camera motion. The experimental task had the helicopter operator hover the vehicle over a location marked on the floor. Once a stable hover was achieved the operator maneuvered the vehicle to hover over a second marked location. The task of moving between the two locations was conducted approximately 20 times. The experiment was executed at slow, medium, and fast speeds for each of the teleoperation configurations.

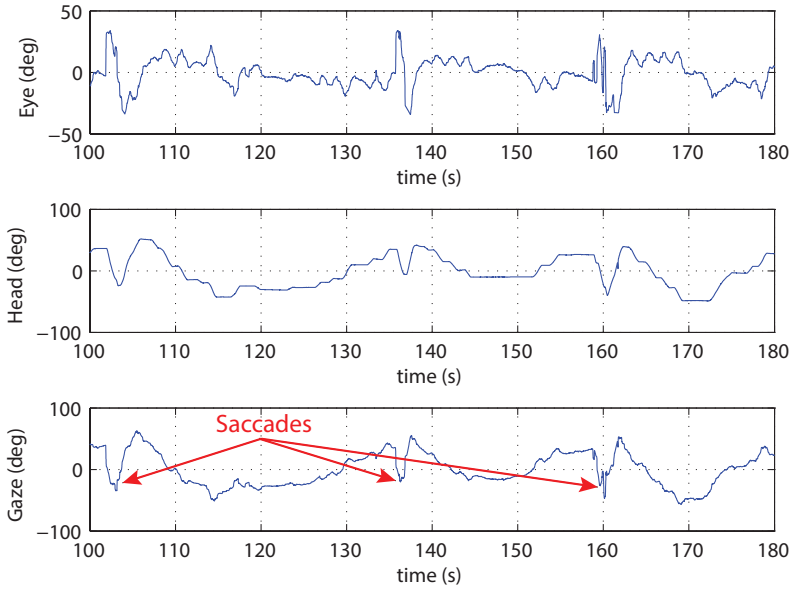
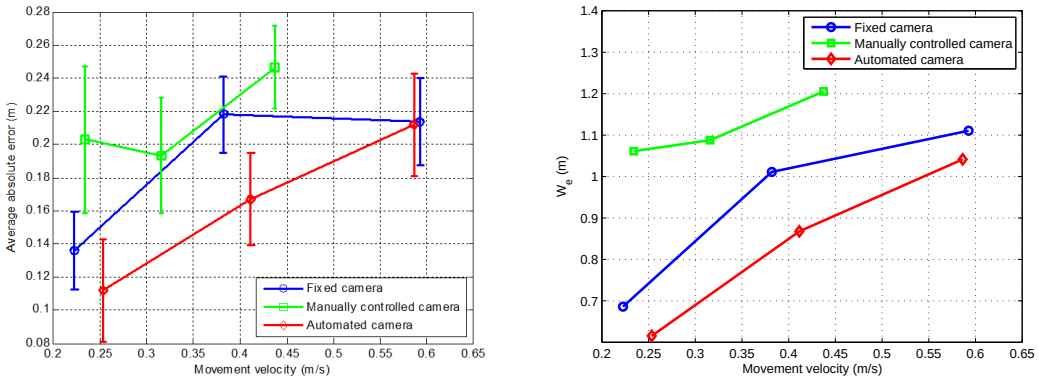


Figure 8.8: Saccades generated by the eye tracking/gimbal system.

The Fitts metric defined in Section 5.3.2 was used to analyze performance. The experiment defined the movement velocity and the teleoperation configuration as the independent variables with W_e as the dependent variable. The expectation was that plotting the movement velocity against W_e for the different configurations would generate approximately linear results with the more challenging teleoperation configurations having higher W_e values. Fig. 8.9 shows the results for the three configurations and three movement speeds. Based on the analysis, the proposed approach was the easiest for the operator while the manually operated camera proved the most difficult. The hypothesis for the narrower range of speed for the manually controlled camera is that the pilot had more difficulty perceiving the helicopter motion than for the other two configurations. The pilot reduced the speed for the medium and fast cases to compensate for impaired perceptual abilities.

Obstacle Avoidance

To demonstrate the capabilities of the proposed system for a more complex task, experiments were conducted using the task configuration shown in Fig. 8.10. During this task, the operator's



(a) Error versus speed comparison.

(b) Effective width versus speed comparison.

Figure 8.9: Comparison of task difficulty for different teleoperation configurations.

head-eye direction would typically start in *smooth pursuit* mode to support the continuous motion of the helicopter. As the helicopter approaches the first obstacle an *anticipatory saccade* is used to identify an appropriate subgoal location. A return saccade moves the gaze back to the helicopter trajectory where the smooth pursuit mode resumes. This basic interaction has been investigated in our related research [12, 38]. The goal of this work is to investigate the use of information from an eye tracking system to generate control signals for positioning an actuated camera and augment the operator control input.

The experiments are organized according to two configurations. In the baseline configuration, the camera is controlled separately by an additional operator. In the test configuration, the camera is controlled based on the operator's visual gaze pattern. The hypothesis is that under the baseline configuration the external camera control compromises the natural head-eye system. In particular, when the human head-eye coordination processes do not operate in their natural form, it is expected that the operator will have more difficulty obtaining quality helicopter positioning and velocity information. It is also expected to increase the mental workload. As a result, there should be a degradation in the helicopter control performance.

Experimental results for the two test cases are shown in Fig. 8.11. The figure shows a subset of the successful trajectories for each case. The first observation is that the trajectories with the automated camera are smoother and have significantly smaller deviations from the expected trajectory (note the looping behavior exhibited in the manual control test). A second observation is that manual camera operation caused a significant increase in collisions with the obstacles and

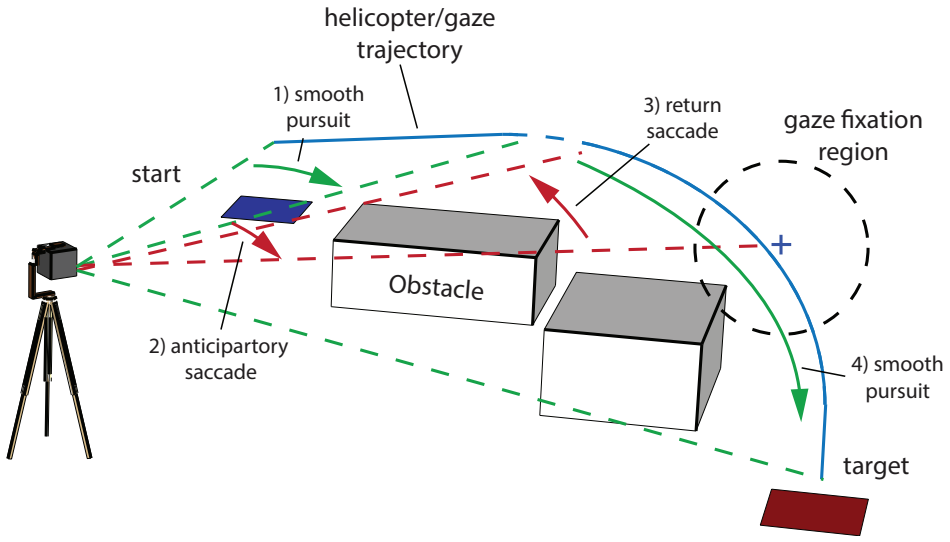
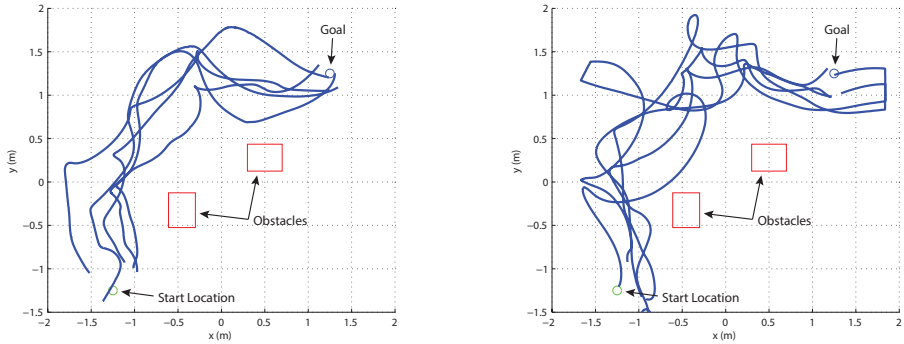


Figure 8.10: Description of gaze pattern for example teleoperation task.

occasionally made it difficult to keep the helicopter in the field of view. A final, subjective, observation was that the automated camera provided a more natural perceptual experience. Since the automated camera mimics the human head, the operator is able to easily incorporate the camera motion into the perceptual process. For manual camera control the camera movement was periodically out of sync with the pilot expectations making the visual optical flow conflict with expected vehicle motion. When the optical flow (due to camera motion) runs counter to the vehicle motion it can be challenging for the human operator to merge the information into a final motion estimate for the vehicle which can lead to poor or even unstable control.

8.4 Augmented Flight Control

A second application example builds on the automated camera gimbal from the previous section. In addition to controlling the camera, processes are implemented to estimate the state of the vehicle and task elements using gaze, generate perceptual guidance trajectories, and perform visual tracking. Experiments using a step task demonstrate the system capabilities. Fig. 8.1 shows the components implemented inside the gray box.



(a) Test flights with automated control of camera motion.

(b) Flight tests with a second operator manually controlling the camera motion.

Figure 8.11: Experimental data for trial investigating benefits of automated control of camera in smooth pursuit mode. The green circles are the starting locations, the red boxes show the obstacle locations, and the blue lines the helicopter trajectories.

8.4.1 Implementation of Augmented Control

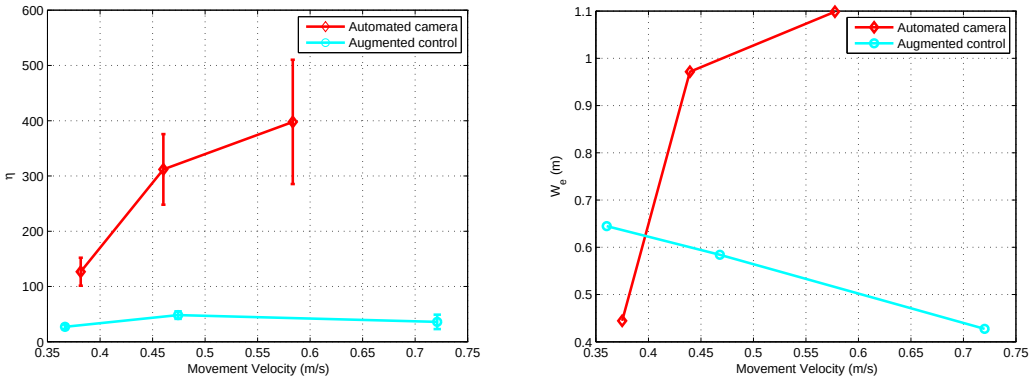
The procedure for automating the camera gimbal was presented in the previous section. The implementation of the remaining components from Fig. 8.1 follows directly from the results of Chapters 6 and 7. The internal model for estimation of the local subgoal and vehicle state was described in Section 7.6. The approach for generating a reference velocity based on the concept of a tau guide is defined in Section 6.4. Finally, the visual tracking component identified in Section 6.2 provided the feedback and feed-forward functions. The only change to the control models was a slight reduction in the feedback gains due to the 120 ms delay in the video feed of the teleoperation system. The individual elements were implemented as real-time processes that were integrated into the ROS software environment described in Section 3.4.

8.4.2 Experimental Results

A set of flight experiments was conducted using the teleoperation system with augmented control. The pilots' were asked to perform the step task from Section 5.2.2. At the beginning of the task the pilot was instructed to achieve a stable hover over the initial location and then signal the start of the motion by toggling a switch on the joystick. The augmented control system would then generate a control signal to move the vehicle to the target location with the pilot

having the capability to correct errors in the trajectory using the joystick lateral and longitudinal control inputs. In general, the majority of the motion was successfully handled by the control augmentation with the pilot only providing minimal corrective action around the target.

Fig. 8.12 shows a comparison between a system configuration that only automated the camera motion and a system with the full control augmentation. The workload metric was defined in Section 5.3.3 and the accuracy (given by effective width W_e) in Section 5.3.2. As seen in Fig. 8.12(a), the augmented control configuration had a significantly lower workload than the automated camera configuration. This indicates that the control augmentation is able to successfully take over a large portion of the control action, thus reducing the workload on the pilot. Fig. 8.12(b) demonstrates that the augmented control system also does well when comparing accuracy. The augmented control configuration has slightly worse accuracy at low speed when compared to the automated camera configuration, but as the speed increases the augmented control clearly performs better.



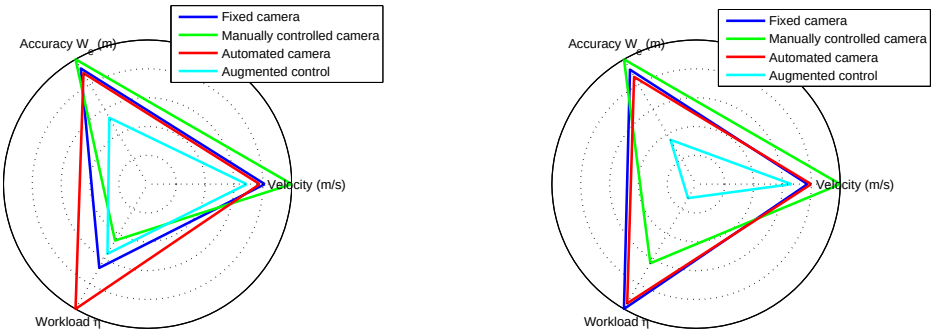
(a) Workload versus speed comparison.

(b) Effective width versus speed comparison.

Figure 8.12: Comparison of operator workload for different teleoperation configurations.

In Chapter 5 a set of metrics for evaluating performance was defined based on "the big three" of speed, accuracy, and workload. Fig. 8.13 shows the metrics on a single plot to make comparison between configurations easier. The configurations tested were a fixed camera, a manually controlled camera (by a second operator), the automated camera of Section 8.3, and the augmented control approach. The Figs. 8.13(a) and 8.13(b) show results for the step task when performed at medium and fast speeds. In the figures lower values in each dimension (accuracy, velocity, and workload) are considered better performance. At medium speed the

augmented control configuration had the best accuracy and velocity and second best workload (behind the manually controlled camera). For fast speed, the augmented control provided the best performance for all three metrics.



(a) Medium Speed.

(b) Fast Speed.

Figure 8.13: Comparison of speed, accuracy, and workload trade off for different teleoperation configurations.

8.5 Conclusions

The chapter described two application examples. The first application automated the positioning of a remote camera based on measurement of the operator's gaze. The camera position control system is based on models of head-eye coordination that are adapted to use eye tracking data as the input. The proposed architecture enables control of the remote camera that mimics human head movement and consequently is more natural for the operator. The second application example augmented the control system to aid the pilot during flight tests.

The proposed automated camera gimbal was evaluated using a version of Fitts Law that showed the system exhibited improved performance in comparison to a fixed or manually operated camera. Experimental results also showed the benefits of the system for a more complex obstacle avoidance task. When compared to the manually operated camera configuration, the proposed system demonstrated smoother trajectories and fewer failures.

The augmented control configuration demonstrated better performance for the accuracy, speed, and workload metrics when compared to the other teleoperation configurations (a fixed

camera, a manually controlled camera, and a camera with automated motion). Subjective comments by the test pilots confirmed the benefits of the augmented control system in that they thought the system made the task easier and was preferable to the other configurations.

Chapter 9

Conclusions and Future Directions

9.1 Conclusions

In this dissertation we presented a general framework to model a human operator's control and guidance response. The framework captures the human-vehicle-environment interactions using modeling techniques based on dynamic systems and controls. A multi-loop architecture organizes the control response into three levels. The lowest level acts as a simple control element to perform visual tracking. The next level, perceptual guidance, employs learned patterns of motion to close gaps between the perceived state and the desired state. The final level considers the task and environment to determine the current subgoal. This dissertation characterized the first two levels of the multi-loop control architecture and identified specific models for the perceptual guidance and visual tracking components. With respect to perception, the visual information extracted by the human operator is registered in a body centric spherical reference frame that corresponds to human visual perception. Decomposition of gaze measurements into smooth pursuit and saccades provided the information necessary to estimate the state of the vehicle and task elements.

The dissertation begins with a description of the hardware and software infrastructure necessary to conduct human-in-the-loop experiments. The lab facility, developed specifically for this work, provides measurements of the human operator's perception-action as well as vehicle motion. The data collected includes human control inputs, visual gaze, and motion tracking for both the human and vehicle. The captured data is synchronized and registered in a common body centric spherical reference frame suitable for analysis.

Next, we identified models for the perceptual and control components of the multi-loop architecture. The pilot's control response was characterized by identifying models for the perceptual guidance and visual tracking blocks. Tau theory provided the basis for modeling perceptual guidance. The concept of a tau guide was utilized to generate reference trajectories by matching the tau guide generated control with the experimental control response. At the visual tracking level, we identified models for feed-forward and feedback control elements. The feed-forward element was found by inverting a simplified version of the helicopter dynamics. System identification techniques fit the control parameters for the feedback component. Gaze was utilized as the primary sensory mechanism for measuring vehicle state and task elements. We demonstrated how the gaze patterns can be decomposed into smooth pursuit and saccades. These visual cues were analyzed to determine the primary visuo-motor control mechanisms in the multi-loop human control system. We showed that the smooth pursuit gaze behavior provides a measure of the rotorcraft state necessary for stabilization and regulation. The saccades, on the other hand, measure the gap to the goal location, which is consistent with tau theory. Using the information extracted from gaze we designed an estimation model that tracked the vehicle state and task elements.

Finally, we applied the models for human perception and control to a real world problem. The first example application utilized gaze to automate positioning control of a remote camera based on models of the human head/eye system. The architecture implements control of the remote camera that mimics human head movement and consequently is more natural for the operator. We evaluated the application using a version of Fitts Law that showed the system exhibited improved performance in comparison to a fixed or manually operated camera. A second example application augmented the control system to aid the pilot while still allowing pilot input. The augmented control example demonstrated better performance for the accuracy, speed, and workload metrics when compared to the other teleoperation configurations (a fixed camera, a manually controlled camera, and a camera with automated motion).

9.2 Future Directions

The research presented in this dissertation focused on a "third-person" perspective, i.e., the operator views both the vehicle and task from outside. The majority of the results would also apply to a "first-person" perspective where the camera is mounted on the vehicle. The primary

difference involves the method employed to measure the vehicle state. An interesting avenue for future investigation would be to identify the human based visual cues necessary to estimate motion when operating in a "first-person" configuration (see Appendix A for an approach based on modern computer vision techniqes)

A second direction for future research involves combining the models presented in this dissertation with work on higher level navigation. Separate research is ongoing at the Interactive Guidance and Control Lab (IGCL) to better understand how subgoals are identified and what triggers a switch to a new subgoal. Integrating the perceptual guidance and visual tracking components from this dissertation with methods for higher level navigation would provide a complete solution for more complex tasks.

Appendix A

A Vision Based Ensemble Approach to Velocity Estimation for Miniature Rotorcraft

This Appendix describes a technique for estimating a vehicle's body velocities and Euler angles that incorporates vision. The goal of the work was to understand how vision can be utilized in a state estimation system. The work done in collaboration with Anoop Cherian proposed a novel framework for state estimation that combines a dynamic flight model, IMU measurements, and 3D velocity estimates computed from an onboard monocular camera using computer vision. The work differs from existing approaches in that, rather than using a single vision algorithm to update the vehicle's state, we capitalize on the strengths of multiple vision algorithms by integrating them into a *meta-algorithm* for 3D motion estimation.

A.1 Introduction

Research efforts utilizing miniature rotorcraft are rapidly expanding from typical outdoor applications such as search and rescue, inspection, surveillance, and reconnaissance to indoor or confined environments. Effective operation of the vehicle requires feedback control for stabilizing the vehicle and damping out disturbances, at the same time allowing the human operator to provide control inputs. This wprl considers low cost miniature rotorcraft that are suited for

operation in indoor environments. These types of miniature rotorcraft present some unique challenges. One significant challenge is the stringent payload limitation that restricts the sensors and processing capabilities that can be mounted on the vehicle. Another challenge arises due to operation in an enclosed environment where sensors such as GPS or magnetometers are unreliable. As a consequence of these challenges, implementing control systems can be a difficult problem.

In order to build a control system, estimates of the vehicle states must be available. In general, low cost IMUs and other small footprint sensors like cameras introduce significant measurement errors, thus making it difficult to base a control strategy on any individual sensor. To compensate for the limitations of the individual sensors, our approach fuses information from the dynamic vehicle model, measurements from an onboard IMU, and a camera in order to estimate the Euler angles (roll, pitch, and yaw) and body frame velocities needed for control. Each component provides different modalities of information and a state estimation approach that fuses them will provide better performance than a subset of the elements.

The onboard camera enables a powerful set of motion cues for velocity estimation. However, most of the vision based motion estimation algorithms utilize only a single motion cue (such as optical flow or large camera displacement), as a result they provide only partial information regarding the motion. Inspired by this observation, we propose a novel ensemble approach in which we fuse together motion estimated from multiple computer vision algorithms to get a robust estimation of the vehicle velocity direction. At this point, we would like to point out that the visual cues cannot, by themselves, determine the 3D depth; as a result they can only provide the direction of the vehicle velocity. Nevertheless, the proportions of the 3D velocity contained in the direction vector provide valuable information. That is, mismatch between the dynamic model and the physical system can result in estimation error that can accumulate and cause drift in the states. The velocity direction allows for correction of low frequency estimation errors as our experiments demonstrate.

Before we proceed, let us enlist the main contributions of this work. We propose an Extended Kalman Filter (EKF) based state estimation framework that combines the vehicle's dynamic flight model, the MEMs IMU readings and vision based estimates of the 3D velocity direction vector from a front facing monocular camera. Next, we introduce a novel vision based motion estimation framework that exploits the strengths of different vision algorithms into a meta-algorithm to determine the vehicles' unit-velocity. We demonstrate the effectiveness of

our method via real implementations on two helicopter platforms: (i) the commercially available AR Drone system, and (ii) a coaxial helicopter platform with more challenging dynamics.

A.2 Related Work

State estimation for aerial vehicles is an established procedure when GPS and high-end IMUs are available. Significant work exists on performing state estimation during temporary drop outs of GPS signals [122], without using the vehicle's dynamic model [123], or using low-cost noisy IMU readings [124]. However, state estimation in true GPS denied environments, such as the problem we investigate in this work, still needs attention.

Estimating the camera ego-motion (3D rotation and translation) is a well studied problem in computer vision and a variety of techniques are available [125]. Motion estimation techniques using an onboard monocular camera generally fall into one of the two broad categories: namely (i) discrete time methods, and (ii) continuous time methods. The former type assumes a significant translation of the camera between a pair of consecutive images and utilizes the geometric constraints (also called *epipolar constraints*) between a static world point and its projection on the images to infer the camera motion [126]. Continuous time methods use the motion induced intensity fields (optical flow) between consecutive images, and use structural constraints of image formation (such as subspace constraints, coupling between optical flow with the vehicle's linear and angular velocities, etc.) to estimate the 3D motion [127]. These vision schemes have been applied to state estimation problems in aerial vehicles before, such as in the work of Soatto et al. [128] that uses the epipolar constraint or in [129, 130] that use flow based subspace constraints.

In a real-world setting, especially for a 6-DOF helicopter, many of these motion estimation algorithms based on monocular vision fail. It is not only the dynamic operating environment that adds to the challenge, but also the influence of external noise such as jitter, radio frequency interference, illumination variations, etc. that cause variability in feature position and false image correspondences. In [131], a Kalman filter based fusion model is introduced, that not only measures the dynamic state of a UAV operating indoors, but also estimates the wind velocity. This system uses a downward looking camera and makes restrictive assumptions on the ground plane. In [132], a sigma-point Kalman filter is proposed that fuses motion estimates from a forward-looking camera, a 3-axis airspeed sensor, and an onboard IMU. They use the epipolar

geometric algorithm, but rely on the airspeed sensor for state estimation. Unfortunately, this sensor requires extensive calibration over the expected flight conditions and is often less applicable to rotorcraft operating at low speeds. In [133], vision aided navigation systems that use low-cost inertial sensors are proposed to operate in environments where the GPS is temporarily denied. The approach is based around an inertial navigation system (INS) and updated based on estimates of the 3D position of tracked features. In our work, the end goal is to use the state estimates as part of a closed loop controller. To integrate all sources of information and ensure a stable response with known dynamics, a dynamic model is used as the central block instead of a simple INS system.

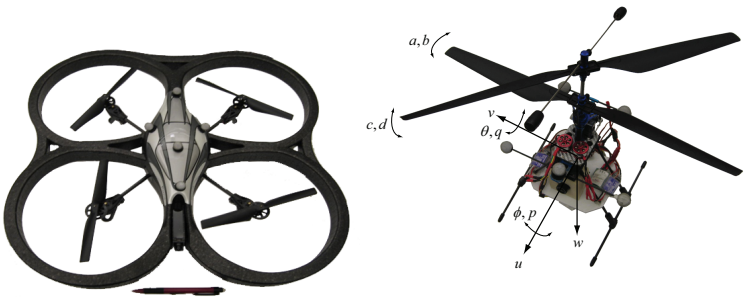
In [134], two cameras are integrated with an IMU to generate state estimates onboard a micro air vehicle (MAV). A state estimation approach for MAVs using a single camera was described in [135]. The approach fuses pose estimates obtained using VSLAM with information from fast sampling inertial sensors. A method of incorporating optic flow as a measurement update for a visual-inertial state estimator was presented in [136]. The existing method that most closely resembles the approach proposed in this work is described in [137]. The work combines a single camera and IMU to estimate velocity and then proceeds to combine the method with VSLAM for navigation. Our approach also uses a single camera and IMU to measure speed, but we utilize a dynamic model of the vehicle and multiple vision algorithms. A few additional approaches closely related to ours are [138, 139, 140] that utilize dynamic models of aerial vehicles combined with vision based measurement updates. All these approaches use only a single vision based motion estimation algorithm. In real world flights, individual algorithms can have problems with specific operating environments and motions. The problem with the individual vision algorithms is that each looks at a particular motion cue for separating the non-linearly coupled motion parameters. To circumvent this problem and at the same time take advantage of the higher diversity of motion cues available in front of the camera, we propose a *meta-algorithm*, that fuses the velocity estimates from a small set of state-of-the-art ego-motion algorithms in an EKF setting. To the best of our knowledge such a meta-algorithm has not been suggested before.

A.3 Experimental Setup

The experimental facility was described in Section 3. This section provides additional details on the helicopter platforms and sensors.

A.3.1 Helicopter Platforms

A number of potential rotorcraft platforms are commercially available in single rotor, coaxial, and quadrotor configurations. Our two main selection criteria were: (i) small size suitable for indoor operation, and (ii) the ability to carry a sensor suite (a payload of at least 50 grams). Other criteria were cost and integrated electronics. One commercial rotorcraft platform that meets these requirements is the AR Drone quadrotor, which is small enough (52.5 x 51.5 cm, 420 grams) and includes all the necessary onboard sensors. The AR Drone also provides onboard processing for stabilization that includes fusion of optical flow from a downward camera with inertial measurements [141]. For this work only the raw measurements from the AR Drone's IMU and camera were used. The AR Drone is shown in Figure A.1(a).



(a) (b)

Figure A.1: Helicopter platforms: (a) the AR Drone quadrotor platform, and (b) the coaxial helicopter platform based on the Blade CX2 showing the state variables defined in the dynamic model.

The primary quantities of interest in this work are the estimated body velocities for the rotorcraft platforms. To demonstrate the effectiveness of our velocity estimation approach for a diverse set of rotorcraft configurations, two platforms with significantly different dynamics were investigated. The two platforms show that the proposed estimation method is applicable to the full range of miniature rotorcraft. The first platform, the AR Drone, provides a simple dynamic

response. The second platform, a coaxial helicopter based on the Blade CX2 from E-Flite[1] , had significantly more complex dynamics. The Blade CX2 has a rotor diameter of 34.5 cm, a height of 18.3 cm, and weighs 220 g (with battery). Figure A.1(b) shows the modified Blade CX2 equipped with a video camera. For more details on this platform, see [91].

A.3.2 Sensors

The test vehicles include a range sensor for altitude measurement, a forward facing camera, and an IMU for motion and pose estimation. The IMU onboard the vehicle provides measurements of the angular rates and accelerations of the body frame. To generate the Euler angles based on the IMU measurements, a complementary filter was applied. The complementary filter is a simple approach that combines angle measurements based on the accelerometers' low frequency response with integrated gyro measurements (that capture the higher frequencies) [142]. Experimental testing shows that the response of the complementary filter accurately matches the actual Euler angles (as provided by the Vicon system).

During circular trajectories the gravitational direction measured by the IMU will be incorrect (due to centripetal force). Consequently, the attitude estimate generated by the complementary filter will be inaccurate during such maneuvers. Improved attitude estimation could have been achieved by incorporation of the roll and pitch angles into the filter framework discussed later in the appendix, but this was deemed unnecessary for the current system because the specific motion that causes the issue was not a significant component of the experimental trials.

Using the Euler angles, the acceleration due to gravity can be removed from the IMU's accelerometer measurements to get the accelerations in the body frame. To improve the IMU measurements, a calibration procedure is applied before each flight test. The resulting measurements of the angles and body velocities are provided as inputs (or updates) for the state estimation algorithm.

A.4 Dynamic Models

This section provides an overview of the dynamic models for the two rotorcraft from Section A.3.1. The coaxial is more complex due to the coupling between the two rotors, flybar, and

[1] http://www.horizonhobby.com

fuselage motion. For the AR Drone and coaxial vehicles, the control inputs are given by:

$$u(k) = \begin{bmatrix} u_{lon}(k) & u_{lat}(k) & u_{col}(k) & u_{ped}(k) \end{bmatrix}^T \tag{A.1}$$

where u_{col} controls the height, u_{ped} the heading, u_{lon} the longitudinal velocity, and u_{lat} the lateral velocity. The basic structure of the algorithm for estimating state information is shown in Figure A.2 where u, v, and w are the body velocities, a_x, a_y, and a_z are body accelerations, and p, q, and r are the angular rates. The state estimation block uses a standard Extended Kalman Filter (EKF).

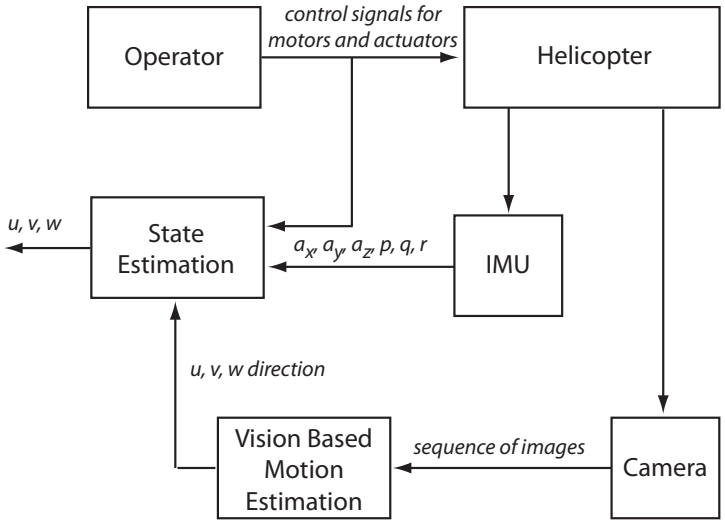


Figure A.2: Block diagram of the state estimation approach.

The structure of the algorithm for state estimation integrates complementary sources of information. The dynamic model provides structure for the state estimates based on real world dynamics while the IMU measurements correct for higher frequency errors. The vision algorithm compensates for low frequency errors caused by modeling error or drift in the states.

A.4.1 AR Drone

For the experimental evaluation described in this work, the onboard control loops on the AR Drone were left in place. The dynamic model of the quadrotor (including the feedback control loops) was identified using frequency sweep data for the longitudinal and lateral control inputs. The model parameters given in Table A.1 were identified by first extracting the frequency

responses and then fitting these with transfer functions using FRESPID and NAVFIT tools in CIFER respectively [94]. The state and noise vectors are given by $\hat{x} = \begin{bmatrix} u & \dot{u} & v & \dot{v} & w & \dot{w} \end{bmatrix}^T$ and $\omega = \begin{bmatrix} \omega_u & \omega_v & \omega_w \end{bmatrix}^T$. The second order transfer function equations from control inputs to vehicle velocities (u, v, w) are given in (A.2). The measurement updates for the state estimator come from the onboard accelerometer, zero velocity updates based on optical flow, and velocity direction updates from the computer vision algorithms.

$$\begin{aligned} \ddot{u} &= U_u u + U_{\dot{u}} \dot{u} + U_{lon} u_{lon} + \omega_u \\ \ddot{v} &= V_v v + V_{\dot{v}} \dot{v} + V_{lat} u_{lat} + \omega_v \\ \ddot{w} &= W_w w + W_{\dot{w}} \dot{w} + W_{col} u_{col} + \omega_w. \end{aligned} \tag{A.2}$$

Parameter	Value	Parameter	Value	Parameter	Value
U_u	-4.471	$U_{\dot{u}}$	-6.567	U_{lon}	24.05
V_v	-5.481	$V_{\dot{v}}$	-6.581	V_{lat}	22.56
W_w	-36.06	$W_{\dot{w}}$	-12.02	W_{col}	-28.34

Table A.1: Model parameters for the AR Drone dynamic model.

A.4.2 Coaxial Helicopter

The dynamic model structure used in this appendix is based on previous work on modeling and characterizing the dynamics of miniature rotorcraft [87, 95]. The helicopter control inputs are measured at each time step. Based on these control signals, one can integrate the helicopter equations of motion forward in time to obtain an estimate of the vehicle's actual state.

The dynamic model identified in [87] provides a set of linearized equations of motion for the helicopter and has the form shown in (A.5). The relation of the state variables to the helicopter body is shown in Figure A.1(b). The numerical values of aerodynamic, control derivatives and model parameters listed in Table A.2 were taken from [87]. The coefficients τ and τ_s are the rotors' and stabilizer bar time constants, respectively. In the model, the stabilizer bar and upper rotor were combined and represented by c and d states (aerodynamic effects like those discussed in [143] are ignored). The state vector shown in (A.3) includes the Euler angles (ϕ, θ, ψ), the angular rates (p, q, r), the flapping angles (a, b, c, d), and the body velocities (u, v, w).

$$\hat{x} = \begin{bmatrix} \phi & \theta & \psi & p & q & r & \dot{r} & a & b & c & d & u & v & w \end{bmatrix}^T \tag{A.3}$$

The noise vector is given by

$$\omega = \begin{bmatrix} \omega_p & \omega_q & \omega_r & \omega_a & \omega_b & \omega_c & \omega_d & \omega_u & \omega_v & \omega_w \end{bmatrix}^T. \tag{A.4}$$

$$\begin{aligned}
&\dot{\phi} = p, \quad \dot{\theta} = q, \quad \dot{\psi} = r, \quad \dot{p} = L_b b + L_d d + \omega_p \\
&\dot{q} = M_a a + M_c c + \omega_q, \quad \ddot{r} = R_{\dot{r}} \dot{r} + R_r r + R_{ped} u_{ped} + \omega_r \\
&\dot{a} = -q - \tfrac{a}{\tau} + A_{lon} u_{lon} + A_{lat} u_{lat} + A_u u + \omega_a \\
&\dot{b} = -p - \tfrac{b}{\tau} + B_{lon} u_{lon} + B_{lat} u_{lat} + B_v v + \omega_b \\
&\dot{c} = -q - \tfrac{c}{\tau_s}, \quad \dot{d} = -p - \tfrac{d}{\tau_s} \\
&\dot{u} = X_u u + X_\theta \theta, \quad \dot{v} = Y_v v + Y_\phi \phi \\
&\dot{w} = Z_w w + Z_{col} u_{col}
\end{aligned} \tag{A.5}$$

Parameter	Value	Parameter	Value	Parameter	Value
L_b	265.4	A_{lat}	-2.908	X_u	-0.446
M_a	189.6	B_v	-2.141	X_θ	-9.810
A_u	2.289	B_{lon}	-2.232	Y_v	-0.604
A_{lon}	-4.157	B_{lat}	4.900	Y_{phi}	9.810
M_c	154.5	L_d	186.2	τ_s	0.559
Z_w	-1.36	Z_col	-10.0	τ	0.028

Table A.2: Miniature helicopter aerodynamic and control derivatives (units are in degrees and meters).

The measurement updates for the state estimator use the Euler angles calculated using the complimentary filter, zero velocity updates, and velocity direction updates from the computer vision algorithms.

A.5 Velocity Direction Estimation Using Computer Vision

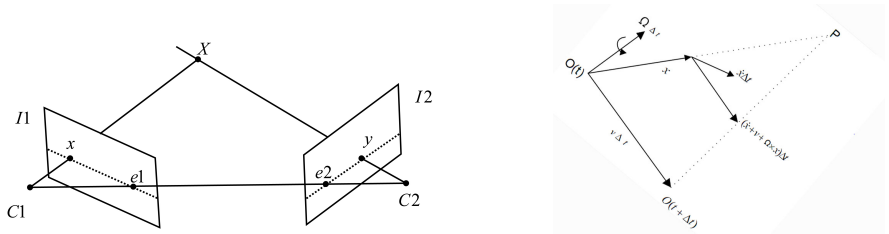
The translational velocity of a helicopter estimated using an IMU alone will drift due to noise and errors in the accelerations resulting from corrections for gravity. Consequently, another measurement modality to update the estimate is essential. Computer vision provides a complementary source of velocity information. Since the scene depth cannot be estimated using a monocular camera, the computer vision algorithms cannot find the magnitude of vehicle velocity, instead a velocity direction vector is identified and used as an observation for state estimation. It is seen that a single vision algorithm might not be sufficient for robust estimation of

the translational velocity given the diversity of the scenes in front of the camera. For example, optical flow based algorithms work best for estimating instantaneous velocities, while epipolar geometry is superior when the camera translation is large. Thus, a fusion of these disparate vision algorithms can capitalize on the strengths of the individual algorithms and provide robust ego-motion estimation. In the following subsections, we will elucidate these vision algorithms in detail, before introducing our velocity fusion framework.

A.5.1 Computing Point Correspondences

The first step in any vision based motion estimation algorithm is to compute point correspondences across successive video frames; these correspondences are later decomposed into the velocity components. Two significant problems need to be addressed at this stage: (i) robustness of the feature descriptors and (ii) real time computation of these descriptors. After experimenting with several popular methods, we selected the recently introduced Binary Robust Independent Elementary Feature (BRIEF) descriptors [144]. These descriptors are computationally efficient, robust and take much less storage (128–512 bits) than other popular methods.

A.5.2 Epipolar Geometry



(a) (b)

Figure A.3: (a) Illustration of the epipolar constraint. The camera centers $C1$ and $C2$, along with the world point X, lie on the same epipolar plane. (b) Schema showing the relation between the 3D velocity components and the optical flow.

One of the most popular vision algorithms for ego-motion estimation is based on epipolar geometry [125]. The algorithm assumes the camera motion is smooth and the environment is stationary. It takes as input the point correspondences computed between two successive video frames. Assuming a pinhole model, suppose the camera center moves from a world point $C1$

to $C2$. Further, let $\mathbf{x}$ and $\mathbf{y}$ represent the homogeneous image coordinates of a stationary world point X projected onto the camera image plane at $C1$ and $C2$ respectively. Then, epipolar geometry says that the camera centers and the world point are coplanar. This coplanarity constraint can be written in terms of two corresponding points $\mathbf{x}$ and $\mathbf{y}$ as:

$$\mathbf{x}^T \mathbf{F_e} \mathbf{y} = 0, \tag{A.6}$$

where $\mathbf{F_e}$ is called the *Fundamental matrix*. It is well-known [126] that $\mathbf{F_e}$ can be written as:

$$\mathbf{F_e} = RS \tag{A.7}$$

where R is the 3×3 rotation matrix of camera motion and S is the skew-symmetric representation of the camera translation vector $T = [t_x, t_y, t_z]^T$, i.e.

$$S = \begin{pmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{pmatrix}. \tag{A.8}$$

Note that, for tractability of the solution, the translation vector is normalized to unity. Given eight or more point correspondences between the two images, we use the Gold Standard algorithm in conjunction with RANSAC to robustly estimate $\mathbf{F_e}$ [125], from which the rotation matrix R and the unit translation vector T can be derived using simple linear algebra [125]. Unfortunately, the signs of the dimensions of T are generally unpredictable in practice (due to errors in computing the point correspondences). We will tackle this problem in Section A.6.2.

Although the epipolar algorithm is computationally simple, it has several disadvantages as listed below:

1. It assumes that there is sufficient translation of the camera, which is not always the case. For example, if the motion is a pure rotation, then epipolar geometry will still produce a unit magnitude translation vector due to the vector normalization and numerical errors.

2. It often uses a RANSAC step for estimating the fundamental matrix robustly, in which multiple motion models are fit to subsets of the point correspondences to make the fit robust to outliers. When there is significant noise in the images, a large number of iterations are required to find good fitting models, demanding high computational load.

Alternative ego-motion estimation algorithms that can help circumvent these fundamental issues will be described in the following subsections. Our primary criteria for selecting these algorithms is to consider different assumptions about the scene in front of the camera, since a single assumption or visual cue might not be sufficient for a real world application of the system.

A.5.3 Optical Flow

Optical flow is one such cue for ego-motion estimation. Unlike the geometric method, optical flow algorithms work on instantaneous (differential) velocities, which means that they are more suited in situations where the motion of the helicopter is small. In this work, we investigate the less commonly used *fundamental equation of optical flow*. Although, optical flow is very popular in computer vision for estimating the image motion across frames, we would like to emphasize that we are dealing with algorithms for computing the 3D camera motion from the 2D optical flow vectors. Towards this end, we chose two potential algorithms: namely (i) the Kanatani's renormalization method and (ii) the linear subspace algorithm. Further, assuming we have reasonably accurate estimates of the vehicle's angular velocity, we propose a simple scheme for 3D unit-velocity estimation method using the fundamental equation of flow, which we call the Maximum Likelihood Estimator (MLE) method on flow. In the following, we briefly review the basics of optical flow, before discussing each of these algorithms in detail.

Given two consecutive images from the onboard camera, optical flow describes the instantaneous motion field of image intensities between the two images as a result of the camera motion [145]. Suppose $I(x,y,t)$ signifies the image intensity at coordinates $\mathbf{x} = (x,y)$ and time t. If this intensity moves to $\mathbf{y} = (x+\delta x, y+\delta y)$ at time $t+\delta t$ due to camera motion, then we have:

$$I(x+\delta x, y+\delta y, t+\delta t) = I(x,y,t). \tag{A.9}$$

Expanding the LHS of (A.9) using Taylor series to a first-order approximation, we have the optical flow constraint:

$$\frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \frac{\partial I}{\partial t} = 0. \tag{A.10}$$

A.5.4 Epipolar Equation of Optical Flow

Assume in the following that V and Ω are the camera linear and angular velocities, and let $\mathbf{I}$ represent a 3×3 identity matrix. With reference to Figure A.3(b), let a world point $O(t)$, having

a corresponding image coordinate represented by $\mathbf{x}$, is moved to the point $O(t+\Delta t)$ as a result of a translation $T = V\Delta t$ and rotation $R = \mathbf{I} + \Omega \times \mathbf{I}\Delta t$. Then, at $O(t+\Delta t)$ the new image point $\mathbf{y}$ is approximately (to the first order) given by:

$$\mathbf{y} = \mathbf{x} + \dot{\mathbf{x}}\Delta t. \tag{A.11}$$

Here, we assume $\dot{\mathbf{x}}$ represents the optical flow at the image coordinate $\mathbf{x}$. Using the epipolar constraint from the previous section, we have:

$$|\mathbf{x}, T, R\mathbf{y}| = 0, \tag{A.12}$$

where (A.12) is a triple product. Substituting for T and $R\mathbf{y}$, we have:

$$|\mathbf{x}, \dot{\mathbf{x}}, V| + (V \times \mathbf{x})^T (\Omega \times \mathbf{x}) = 0, \tag{A.13}$$

which can further be written as:

$$\begin{aligned} &\mathrm{Tr}\left[(\dot{\mathbf{x}}\mathbf{x}^T)(V \times \mathbf{I})\right] + \\ &\quad \mathrm{Tr}\left[(\mathbf{x}\mathbf{x}^T)\left[\left(V^T\Omega\right)\mathbf{I} - \frac{1}{2}\left(V\Omega^T + \Omega V^T\right)\right]\right] = 0. \end{aligned} \tag{A.14}$$

Following [146], separating the known and unknown variables in (A.14); let

$$\mathbf{M} = \mathbf{x}\mathbf{x}^T + \dot{\mathbf{x}}\mathbf{x}^T - \mathbf{x}\dot{\mathbf{x}}^T \tag{A.15}$$

$$\mathbf{F_o} = (V^T\Omega)\mathbf{I} - \frac{1}{2}(V\Omega^T + \Omega V^T) + V \times \mathbf{I}, \tag{A.16}$$

then (A.14) can be rewritten as:

$$\mathrm{Tr}(\mathbf{M}^T\mathbf{F_o}) = 0. \tag{A.17}$$

Here $\mathbf{F_o}$ is called the *flow matrix*. Assuming the velocity vector is normalized to unity and under suitable decomposability conditions, it can be shown that the linear and angular velocities can be estimated separately from (A.17) (see p.329 [146] for the proof).

A.5.5 Maximum Likelihood Estimator Method on Flow

When the vehicle's angular velocity is known, we may estimate the unit-translation vector with better accuracy. In this section, we propose a method to achieve this. Our scheme is a variant of prior methods proposed in p.94 [147] and [136, 137], but differs in that we use a maximum

likelihood estimation for computing the unit velocity, while the prior methods directly solve the epipolar equation for the two unknowns in the 3D velocity vector. Assuming homogeneous image coordinates given by $\mathbf{x} = [x\ y\ 1]^T$ and the corresponding flow vectors by $\dot{\mathbf{x}} = [\dot{x}\ \dot{y}\ 0]^T$, if $\Omega = [\Omega_x\ \Omega_y\ \Omega_z]^T$ and the linear velocity be V, then we have the following simplification to (A.12):

$$\underset{V}{\operatorname{argmin}}\ \mathbf{A}^T V \text{ subject to } \|V\| = 1, \tag{A.18}$$

where $\mathbf{A}$ is a $3 \times N$ matrix generated from N optical flow vectors, such that the i-th column of $\mathbf{A}$ denoted A_i will have values

$$A_i = \begin{bmatrix} -(y_i + \Omega_y) \\ \dot{x}_i + \Omega_x - x_i\Omega_z \\ x_i(\dot{y}_i + \Omega_y) - y_i(\dot{x}_i + \Omega_x) \end{bmatrix}. \tag{A.19}$$

The components of A_i are obtained by expanding and collecting the terms in (A.13) corresponding to the i-th image co-ordinates and its respective optical flow. The formulation in (A.18) can be solved using an SVD decomposition on $\mathbf{A}^T$ and choosing the right singular vector corresponding to the smallest singular value as V.

A.5.6 Linear Subspace Method

A drawback of the above methods is that a large set of optical flow vectors are required for the statistical estimation of the noise model and the corresponding renormalization, which might not always be feasible in practice. To accommodate for this situation, we apply a *subspace* method in this work. This technique [127] is based on the simple observation that there exists a bilinear relationship between the unknown depth of the 3D points and the motion parameters, such that each of them can be estimated by fixing the others, there-by solving each unknown variable independently.

The method has been shown to be quite robust to random errors in the flow computation and does not assume smoothness or planarity in the scene. The basic idea of the algorithm is as follows: using a pin-hole camera model with the camera focal length f, an image point $\mathbf{x} = [x, y]^T$ corresponding to a world point $P = (X, Y, Z)$ is related by

$$x = \frac{fX}{Z}; \quad y = \frac{fY}{Z}. \tag{A.20}$$

Assuming the camera moves in a static scene, and using the notation for the linear and angular velocities from the previous sections, the motion of the point P can be expressed as:

$$\dot{P} = -(\Omega \times P + V). \tag{A.21}$$

Now, taking the derivative of (A.20) and substituting in (A.21), we have

$$\dot{\mathbf{x}} = (1/Z)A(x,y)V + B(x,y)\Omega \tag{A.22}$$

where $A(x,y) = \begin{bmatrix} -f & 0 & x \\ 0 & -f & y \end{bmatrix}$ and

$$B(x,y) = \begin{bmatrix} (xy)/f & -(f+x^2/f) & y \\ (f+y^2/f) & -(xy)/f & -x \end{bmatrix}. \tag{A.23}$$

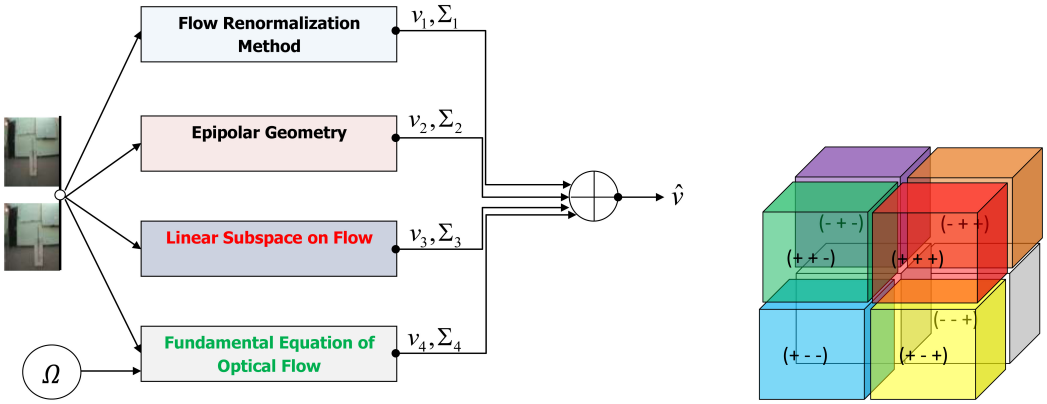
It can be shown that by a rearrangement of terms, (A.22) can be rewritten as

$$\dot{\mathbf{x}} = C(V)\xi \tag{A.24}$$

where $C(V)$ is a matrix with the matrices $A(V)$ and B as columns for all the image coordinates at which the flow is computed, and ξ is a vector of unknowns corresponding to the rotation and the depth variables. This is a bilinear problem with variables grouped in such a manner that it can be optimized separately. In [127], a simple least squares solution is proposed on a discretized solution space that minimizes the error residual between the optical flow vectors and the estimated variables. This algorithm was seen to be robust to noise, works for sparse flow, and at the same time is computationally fast.

A.5.7 Stationary Hover

Another method to estimate the motion is to detect stationary hover. That is, if the average of the absolute flow in the image coordinates in both the horizontal and vertical directions is below a small threshold, the rotorcraft is assumed to be in a stable hover with the velocity vector estimated as zero. This allows for measurement updates directly to the velocity states and helps to correct for drift or model errors.



(a) (b)

Figure A.4: A.4(a) shows the fusion of various vision algorithms for an improved translational velocity estimate. The fusion is implemented using a covariance intersection filter. A.4(b) shows the cuboid histogram for estimating sign of the velocity vector.

A.6 Ensemble Approach to Robust Motion Estimation

As we alluded to earlier, individual vision algorithms utilize disparate motion cues for 3D unit-velocity estimation. For each new image captured from the onboard camera, point correspondences are found between the current and the previous image with the result input to each of the vision algorithms. All the valid outputs from the algorithms are then fused together. In this section, we propose a novel framework for fusing all the vision algorithms for robust estimation. Figure A.4(a) provides an overview of our framework. We assume that each algorithm outputs a 3×3 covariance matrix quantifying its confidence of the unit-velocity estimate. We use the respective velocity mean and this sample covariance for fusion. Since each algorithm uses the same data (scene) in front of the camera, each estimate cannot be considered independent measurements and therefore cannot be used directly in the Kalman filter. Instead we first fuse the estimates using a standard covariance intersection filter [148]. The velocity is subsequently updated within the Kalman filter based on the fused velocity direction. The covariance intersection filter performs the fusion of the covariance and velocity using

$$\Sigma^{-1} = \sum_{n=1}^{N} \lambda_n \Sigma_n^{-1}; \quad \Sigma^{-1}\hat{v} = \sum_{n=1}^{N} \lambda_n \Sigma_n^{-1} \hat{v}_n \tag{A.25}$$

with

$$\lambda_n = \frac{1/\varepsilon_n}{\sum_{i=1}^{N} 1/\varepsilon_i}; \quad \varepsilon_n = \mathrm{tr}(\Sigma_n) \tag{A.26}$$

where $\hat{v}_n$ and Σ_n are the velocity direction estimates and covariances from the different vision algorithms, while $\hat{v}$ and Σ represent the fused velocity and its covariance.

There are a few important practical challenges that need to be addressed in our fusion framework. Note that the unit-velocity measurement from each vision algorithm is weighted by its confidence covariance matrix. Unfortunately, we do not get this matrix directly from the vision algorithms. A second issue regards the sign of the velocity vector. Due to noise, numerical errors, and rank deficiency of the fundamental matrix, the sign of the unit-velocity vector cannot be determined reliably. In the following subsections, we provide statistical heuristics to ameliorate these deficiencies. The final unit-velocity resulting from the fusion framework is validated by comparing it to the predicted observation from the state estimator and discarded if it falls outside a threshold ellipsoid defined by the eigenvalues of the inverse innovation covariance.

A.6.1 Covariance Estimation Via Monte Carlo Simulations

To estimate the confidence covariance of each of the motion estimation algorithms we utilize Monte Carlo simulations. Assume that we have a large set Q of point correspondences, such that $|Q| = N$. Define random subsets $Q^j \subset Q, j = 1,2,\cdots,L$ such that $|Q^j| = m$, where m is heuristically set to double the number of points required by a typical 3D velocity estimation algorithm (such as 16 in case of epipolar geometry). Next, we feed the point correspondences in Q^j for each algorithm $\mathscr{A}_n, n = \{1,2,3,4\}$ and gather the respective velocity estimates. Let the estimates be denoted as v_n^j corresponding to the n-th algorithm and j-th sample set. Then we define the sample mean $\hat{v}_n$ and sample covariance Σ_n of the n-th algorithm as:

$$\hat{v}_n = \frac{1}{L}\sum_{j=1}^{L} |v_n^j|, \; \Sigma_n = \frac{1}{L}\sum_{j=1}^{L} \left[|v_n^j| - \hat{v}_n\right]\left[|v_n^j| - \hat{v}_n\right]^T. \tag{A.27}$$

As we pointed out above, the sign of the velocity vector might not be reliable, as a result we use only the absolute-magnitude.

A.6.2 Statistical Estimation of Velocity Sign

The second important problem, when fusing multiple unit-velocity estimates, is the determination of the sign of the individual 3D directions. The main difficulty is that all the vision methods

are based on the SVD of a rank-deficient matrix, in which the solution vector is identified as the right singular vector corresponding to the smallest singular value. In the presence of noise on the matched features, this singular value fluctuates around zero. Since even the smallest numerical error can cause a change in the sign of the respective singular vector, this estimate is not reliable. Assuming the signs are corrupted by uniformly distributed noise, we employ statistical noise cancellation methods.

Towards this end, we propose the following maximal voting scheme. The signs of unit-velocity vectors in each Monte Carlo simulation is used to vote on a bin in a $2 \times 2 \times 2$ accumulator cuboid, where each bin of the cuboid corresponds to a binary triplet as shown in the Figure A.4(b). The signs corresponding to the largest accumulator bin is used as the direction estimate.

A.7 Experimental Validation

This section starts with simulation based analysis and validation of the performance of the vision algorithms proposed in this work. Following, results from two real-world experiments are presented: (i) for the AR Drone, and (ii) for a coaxial helicopter. The example applications demonstrate the benefits of the vision based ensemble approach. In future work, the approach can be integrated into more complex state estimators.

The vision processing and estimation algorithms were implemented on a basestation to produce velocity estimates in near real-time. The basestation utilized an Intel Core 2 Quad CPU Q9300 running at 2.5 GHz. It should be noted that the computing resources were not computationally constrained. The goal of this appendix is to investigate how a vision based ensemble approach improves velocity estimation and did not focus on optimizing computational requirements.

A.7.1 Simulation Testing of Vision Algorithms

To better understand the performance of the vision algorithms, experiments using simulated data under various conditions were conducted. The five properties of the algorithms we decided to study were: (i) the number of point correspondences per frame pair, (ii) increasing translation of the camera in X direction (in the body frame) while keeping the translations in Y and Z directions fixed, (iii) increasing translation of the camera in Z while translations in the

other two directions fixed, (iv) increasing angular velocity along a fixed axis, while keeping the true translations fixed, and (v) increasing the per-pixel noise for fixed angular and translational velocities. These five experiments reflect our primary motivation to come up with the meta-algorithm for this problem. Results for each experiment are shown in Figures A.5(a)- A.5(e) respectively. Finally, in Figure A.5(f), we show the processing time taken by each algorithm to compute the translation against an increasing number of feature points. As a baseline, all the experiments generated a random set of 1K 3D points, a camera projection matrix using a known translation and rotation was simulated and used to capture 40 corresponding point pairs that appear in the field of view of both cameras. The resulting feature points are then fed into each algorithm. Noise of 2 pixel standard deviation was added to the matching image points to simulate a real-world setting. The error was computed between the estimated unit translation vector and the true normalized translation for 1K trials for each algorithm.

As is clear from the plots, the MLE method is quite robust to noise compared to other methods and performs best for motions in the X and Z directions alone, but it is computationally expensive when compared to other methods due to the SVD step. Surprisingly, the performance of MLE drops as the angular velocity goes up. As expected, the epipolar method works best when the camera motion is significant in the X or Y directions, while the linear subspace is reliable when the motion is large along the Z axis. Overall, we found that the Kanatani renormalization method is the most stable among all the methods we tried, and it was seen to be the most reliable when estimating the sign of the components in the estimated translation. In short, these experiments demonstrate how complementary the algorithms are and motivate fusion of multiple algorithms to achieve robust state estimation.

A.7.2 Experiments with Simulated Test Flights

A near photo-realistic simulator provided an environment for experimental testing of vision updates under scenarios outside the scope of the physical lab [98]. The simulator utilized the dynamics of the coaxial helicopter described in Section A.4.2 and generated images from a simulated camera rigidly mounted to the vehicle. The pilot maneuvered the helicopter down a long hallway while attempting to keep a constant velocity and then returned to the starting area. Since the vision based updates only use the direction component of the velocity, a potential problem exists when operating with a uniform constant velocity. Under constant velocity the vision updates would be unable to correct offset errors. However, in practice, even when a

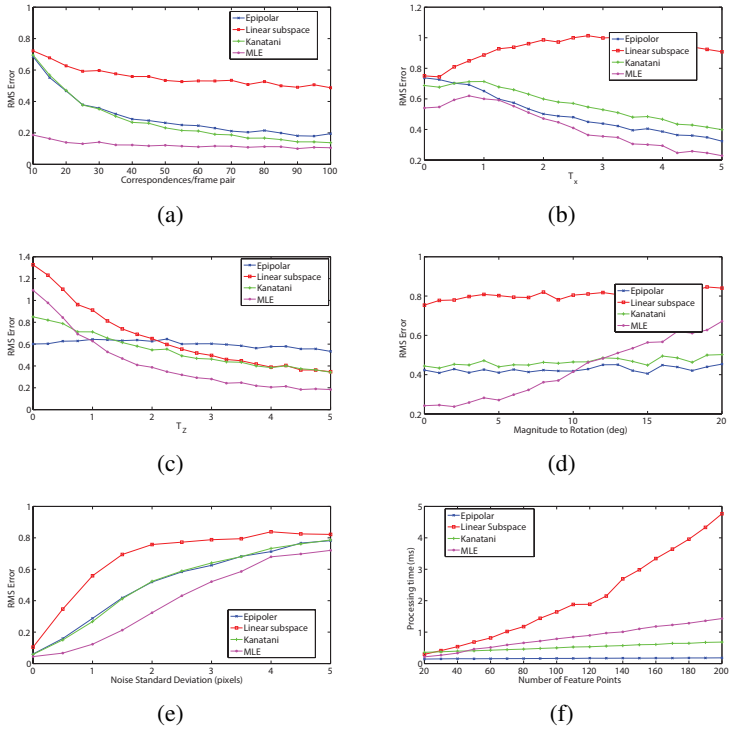


Figure A.5: Plots show the RMSE of each vision algorithm for: A.5(a) increasing number of point correspondences per frame pair, A.5(b) increasing motion only in X direction, A.5(c) increasing motion in Z direction, A.5(d) increasing rotation with fixed translation and A.5(e) increasing image pixel noise. Plot A.5(f) compares the processing time against increasing number of corresponding points to be processed.

pilot attempts to keep a constant velocity there are variations. These variations allow the vision based estimation system to track the velocities accurately over a long period of near constant velocity as shown in Figure A.6(a). Another observation from Figure A.6(a) is that there is a slight decrease in the estimation performance during the middle of the trial where the helicopter performed a rotation to reverse the direction of travel. This is due to few point correspondences being identified during the time of high rotational velocity resulting in a limited number of vision updates.

The vision based updates can be visualized by comparing the directional components with the directional components of the actual velocity. Figure A.6(c) shows the vision updates after fusion of the different algorithms. The vision updates follow the actual values with some level

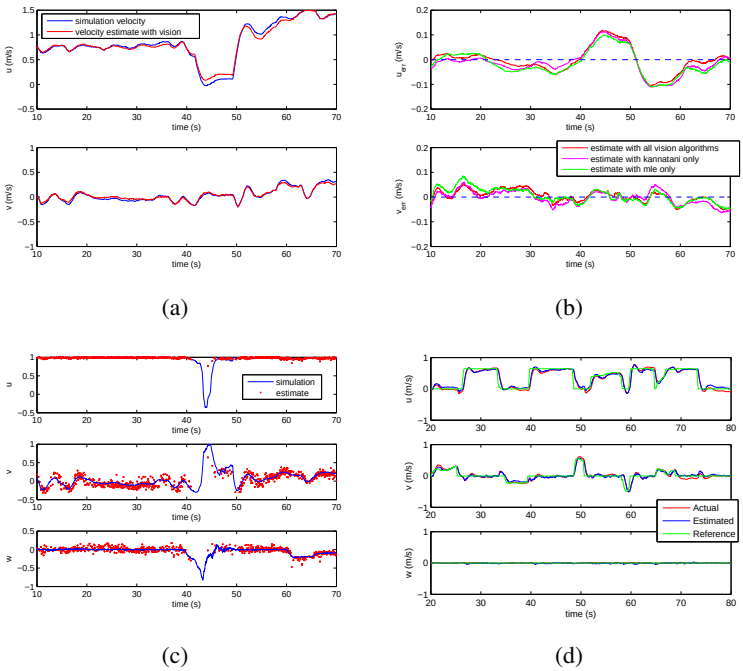


Figure A.6: Simulated flight results: A.6(a) shows the estimated and actual velocities u and v while flying down a hallway, A.6(b) shows estimates for u_{err} and v_{err} when using single vision algorithms compared with the fusion approach, A.6(c) shows the directional components of the vision based updates, and A.6(d) plots the simulated closed loop velocity control showing the estimated, actual, and desired velocities.

of variance. The variance results from noise added to the simulation in order to approximate real performance.

To demonstrate the benefits of fusing multiple vision algorithms a simulated experiment was conducted using each of the vision algorithms independently. The estimation error was calculated for each algorithm and then compared with the results from the fusion. Figure A.6(b) shows the estimated velocities when using two of the vision algorithms independently along with the velocity estimates from the fusion. Table A.3 provides the statistical performance evaluation. As seen in the table, the velocity estimation using the fusion shows improved performance over any of the individual algorithms, providing evidence that the individual algorithms provide different information that may be complementary. To compare to similar state-of-the-art results, the simulated velocity estimation can be compared to the approach in [137]. The

approach also uses a single camera and an IMU, but it only incorporates a single vision algorithm. The estimates for the simulated approach in [137] had an RMS error for the velocity (u,v) of [0.044, 0.050] m/s. Our approach has a similar performance when using only the epipolar algorithm (the one most similar to [137]) with an RMS error of [0.042, 0.044] m/s. When using the fused vision algorithms, our RMS error improved to [0.026, 0.031] m/s.

Vision Update	**RMS error (m/s)**
Epipolar	0.0429
Linear Subspace	0.1052
Kanatani	0.0628
MLE	0.0970
Fused	0.0284

Table A.3: RMS error (compares vectors of estimated velocities with actual velocities) for individual and fused vision algorithms.

To further establish the capabilities of the proposed approach, additional experiments were conducted using the velocity estimates for closed loop control. Control loops were implemented to maintain the desired velocity for the simulated helicopter. The pilot provided the reference velocity using a joystick. Figure A.6(d) shows the estimated, actual, and the desired reference velocities.

A.7.3 Real-world Experiments

In this section, we showcase experimental results after implementing our state estimation algorithm for the two helicopter platforms, namely (i) an AR Drone and (ii) a co-axial helicopter platform. Refer to Section A.3 for details regarding these platforms and the experimental setup.

In our real-world experiments, we found that the number of point correspondences generally fell between 20 and 40. Each vision algorithm was run 30 times on sub-samples of these point sets for the Monte Carlo simulations and we could still achieve 20 frames per second processing performance for the entire vision module.

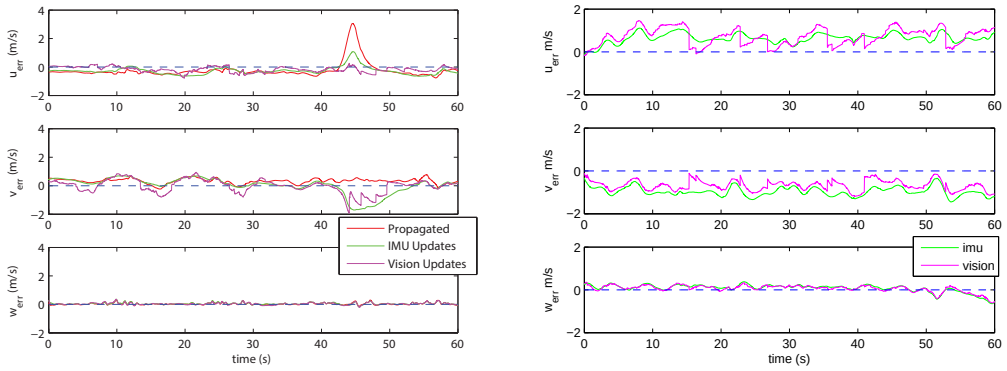
AR Drone: To test the rotorcraft state estimation performance an AR Drone quadrotor was flown in a lab setting. Figure A.7(a) shows the results from the flight test comparing the cases of the dynamic model by itself (propagated), after applying measurement updates from the IMU, and after applying vision based updates. Table A.4 shows the statistical improvement obtained

by applying the vision updates. The values shown are the square root of the MSE found by comparing the vectors of estimated velocities with the velocities measured by the Vicon system (using the goodnessOfFit function in Matlab).

Experiment	RMS error (m/s)
Propagation	0.5044
IMU Update	0.4845
Vision Update	0.3845

Table A.4: RMS error (compares vectors of estimated velocities with actual velocities) for the AR Drone.

Coaxial Helicopter: The coaxial helicopter was used as a representative example for rotorcraft with more complex dynamics. Figure A.7(b) shows the state estimation performance when combining an IMU, camera, and dynamics. Table A.5 shows the statistics for the improvement obtained by applying the vision based updates.



(a) (b)

Figure A.7: Figures A.7(a) and A.7(b) show velocities u_{err}, v_{err}, and w_{err} estimated for an AR Drone and a coaxial helicopter respectively.

Experiment	RMS error (m/s)
IMU Update	0.8456
Vision Update	0.7667

Table A.5: RMS error (compares vectors of estimated velocities with actual velocities) for the coaxial platform.

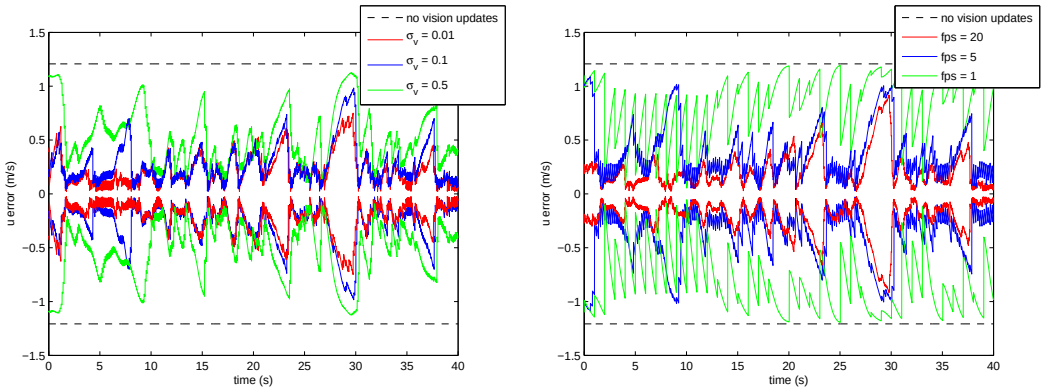
A.7.4 Discussion of Results

The AR Drone experiment looked at the scenario where error in the input signals (due to poorly trimmed control inputs) resulted in an offset to the velocity estimates generated by the dynamic model. The experimental results in Table A.4 show that the vision updates can help correct these offsets. The forward error (u_{err}) after vision updates, shown in Figure A.7(a), has an error magnitude that largely stays less than 0.3 m/s with a slight negative bias. In the lateral direction (v_{err}), the error magnitude stays less than 0.5 m/s with little bias. The magnitude of the error is largely due to low resolution of the captured images (320x240 pixels) while the bias results from the poorly trimmed control signals. A disturbance followed by a short loss of communication (no control or IMU updates, but video still transmitted) occurs around 42 s. As a result, the propagated u_{err} and the v_{err} after IMU updates grow large. This demonstrates that the propagated model and IMU updates by themselves can lead to poor velocity estimates during communication dropouts. If the video is still available, vision based updates can help correct for the velocity errors as seen in Figure A.7(a).

The coaxial experimental result shows how the approach performs for a vehicle with significantly more complex dynamics. In the experiment, the Euler angles that result in a hover state include a bias, which causes an offset in the velocity estimates produced by the dynamic model. This problem is universal for rotorcraft models based on a steady-state hover condition. Changes to the location of the center of gravity also changes the bias of the angles. As seen in Figure A.7(b), the vision updates attempt to move the velocity estimates back towards the actual velocity, but the frequency of the updates is not sufficient to keep the estimates from drifting between the vision based updates. The low frequency of the vision updates, and corresponding large estimation error, is due to noise on the captured images resulting in not enough point correspondences to run the vision algorithms. The images are transmitted using a low power analog wireless video transmitter operating in the 5.8 GHz range with interference causing a significant amount of noise on the captured images. In addition, vibration of the camera during operation causes slight blurring of the images making identification of point correspondences challenging. Nevertheless, after applying the vision updates the velocity estimates are improved as seen in Table A.5. This indicates that the approach can also help compensate for modeling errors if the measurements are available at a sufficiently high rate. In general, we observe that vision updates help correct low frequency errors in the states, while the IMU corrects medium to high frequency errors. The dynamic model, the IMU updates and vision provide complementary

information for state estimation.

The error in the velocity estimates changes based on the update frequency of the vision algorithms and the accuracy of the velocity direction measurements. Figure A.8(a) shows how the 3σ error bounds change as the level of noise on the vision based measurements increases. The σ_v value is the standard deviation of the noise added to each element of the velocity direction vector. Figure A.8(b) demonstrates the importance of the frequency of the measurement updates provided by the vision algorithms. When the frequency drops to 1 update per second, the error bounds become large.



(a) (b)

Figure A.8: Figures A.8(a) and A.8(b) show the 3σ error bounds for the velocity u under changing levels of noise on the vision based velocity direction measurements and changing update rates (frames per second).

A.8 Conclusions

This appendix describes a novel state estimation framework for miniature rotorcraft based on the Extended Kalman Filter. The system combines the dynamic model of the vehicle with IMU measurements and vision-based unit-velocity estimates. To leverage upon the strengths of multiple visual cues, we propose a novel ensemble approach to 3D unit-velocity estimation that fuses together multiple state-of-the-art vision based motion estimation algorithms. The synergy achieved by this combination is experimentally validated in simulations, as well as, real-world flight settings for two helicopter platforms, namely the AR Drone and a coaxial helicopter.

References

[1] Paul M Fitts. Human engineering for an effective air-navigation and traffic-control system. 1951.

[2] DM Pool, PM T. Zaal, MM Van Paassen, and M Mulder. Identification of multimodal pilot models using ramp target and multisine disturbance signals. *Journal of guidance, control, and dynamics*, 34(1):86–97, 2011.

[3] Ronald A. Hess. Model for human use of motion cues in vehicular control. *Journal of Guidance, Control, and Dynamics*, 13(3):476–482, May 1990.

[4] Gareth Padfield and Mark White. Measuring simulation fidelity through an adaptive pilot model. *Aerospace Science and Technology*, 9(5):400–408, 2005.

[5] Hieronymus HLM Goossens and AJ Van Opstal. Human eye-head coordination in two dimensions under different sensorimotor conditions. *Experimental Brain Research*, 114(3):542–560, 1997.

[6] Stephen G Lisberger, EJ Morris, and L Tychsen. Visual motion processing and sensory-motor integration for smooth pursuit eye movements. *Annual review of neuroscience*, 10(1):97–129, 1987.

[7] Adam Fisher. Inside google's quest to popularize self-driving cars. *Popular Science*, 18:2013–09, 2013.

[8] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, K. Lau, C. Oakley, M. Palatucci, V. Pratt, P. Stang, S. Strohband, C. Dupont, L.-E. Jendrossek, C. Koelen, C. Markey, C. Rummel, J. van Niekerk,

E. Jensen, P. Alessandrini, G. Bradski, B. Davies, S. Ettinger, A. Kaehler, A. Nefian, and P. Mahoney. Winning the darpa grand challenge. *Journal of Field Robotics*, 2006. accepted for publication.

[9] Russell H Taylor, Arianna Menciassi, Gabor Fichtinger, and Paolo Dario. Medical robotics and computer-integrated surgery. In *Springer handbook of robotics*, pages 1199–1222. Springer, 2008.

[10] Dario Floreano and Robert J Wood. Science, technology and the future of small autonomous drones. *Nature*, 521(7553):460–466, 2015.

[11] Mica Endsley and Daniel Garland. *Situation Awareness Analysis and Measurment*. CRC Press, 2000.

[12] Jonathan Andersh, Bin Li, and Bérénice Mettler. Modeling visuo-motor control and guidance functions in remote-control operation. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4368–4374. IEEE, 2014.

[13] Edward N Bachelder and Duane McRuer. Perception-based synthetic cueing for night vision device rotorcraft hover operations. In *AeroSense 2002*, pages 377–388. International Society for Optics and Photonics, 2002.

[14] Zhaodan Kong, Bernie Mettler, and Jonathan Andersh. Analysis of pilots guidance behavior based on perceptual guidance principles. submitted to the 2011 IEEE International Conference on Systems, Man, and Cybernetics, 2011.

[15] Stuart K. Card, Allen Newell, and Thomas P. Moran. *The Psychology of Human-Computer Interaction*. L. Erlbaum Associates Inc., Hillsdale, NJ, USA, 1983.

[16] William Powers. *Behavior: The Control of Perception*. Aldine Publishing Company, 1973.

[17] Dietrich Dörner and Harald Schaub. Errors in planning and decision-making and the nature of human information processing. *Applied Psychology*, 43(4):433–453, 1994.

[18] J. Albus, A. Meystel, and S. Uzzaman. Multiscale motion planning. In *Proceedings of the 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation*, 1997.

[19] Marc Lipinski, David R Parks, Robert V Rouse, and Leonard A Herzenberg. Human trophoblast cell-surface antigens defined by monoclonal antibodies. *Proceedings of the National Academy of Sciences*, 78(8):5147–5150, 1981.

[20] Richard W Pew. Human perceptual-motor performance. Technical report, DTIC Document, 1974.

[21] James J. Gibson. *The Ecological Approach to Visual Perception*. Taylor & Francis Group, LLC, 270 Madison Avenue, New York, NY I 0016, 1986.

[22] James J. Gibson. The theory of affordances. In Robert Shaw and John Bransford, editors, *Perceiving, Acting, and Knowing*. 1977.

[23] GJ Pepping and FX Li. Changing action capabilities and the perception of affordances. *Journal of Human Movement Studies*, 39(2):115, 2000.

[24] William H Warren Jr and Suzanne Whang. Visual guidance of walking through apertures: body-scaled information for affordances. *Journal of Experimental Psychology: Human Perception and Performance*, 13(3):371, 1987.

[25] Brett R Fajen and Jonathan S Matthis. Direct perception of action-scaled affordances: the shrinking gap problem. *Journal of Experimental Psychology: Human Perception and Performance*, 37(5):1442, 2011.

[26] Brett R Fajen. Affordance-based control of visually guided action. *Ecological Psychology*, 19(4):383–410, 2007.

[27] William H Warren and Brett R Fajen. From optic flow to laws of control. In *Optic flow and beyond*, pages 307–337. Springer, 2004.

[28] William H. Warren. The dynamics of perception and action. *Psychological Review*, 113(2):358–389, 2006.

[29] Patrick Foo, JA Kelso, and Gonzalo C De Guzman. Functional stabilization of unstable fixed points: human pole balancing using time-to-balance information. *Journal of Experimental Psychology: Human Perception and Performance*, 26(4):1281, 2000.

[30] Dagmar Sternad, Marcos Duarte, Hiromu Katsumata, and Stefan Schaal. Bouncing a ball: tuning into dynamic stability. *Journal of Experimental Psychology: Human Perception and Performance*, 27(5):1163, 2001.

[31] I Siegler, B Mantel, WH Warren, and B Bardy. Behavioral dynamics of a rhythmic ball-bouncing task. In *IV progress in motor control conference, Caen, France*, 2003.

[32] Brett R Fajen and William H Warren. Behavioral dynamics of intercepting a moving target. *Experimental Brain Research*, 180(2):303–319, 2007.

[33] Jack M Loomis, Andrew C Beall, Kristen L Macuga, Jonathan W Kelly, and Roy S Smith. Visual control of action without retinal optic flow. *Psychological Science*, 17(3):214–221, 2006.

[34] Hugo Bruggeman, Wendy Zosh, and William H Warren. Optic flow drives human visuo-locomotor adaptation. *Current biology*, 17(23):2035–2040, 2007.

[35] William H Warren and Brett R Fajen. Behavioral dynamics of visually guided locomotion. In *Coordination: neural, behavioral and social dynamics*, pages 45–75. Springer, 2008.

[36] M.K. McBeath, D.M. Shaffer, and M.K. Kaiser. How baseball outfielders determine where to run to catch fly balls. *Science*, 268(5210):569, 1995.

[37] B. Mettler. Structure and organizational principles of agile behavior: Challenges and opportunities in cognitive engineering. *Journal of Cognitive Critique*, 3, 2011.

[38] Z. Kong and B. Mettler. Modeling human guidance behavior based on patterns in agent-environment interactions. *IEEE Transaction on Human-Machine Systems*, to appear in 2013.

[39] B. Mettler and Z. Kong. Hierarchical model of human guidance performance based on interaction patterns in behavior. London, UK, 2012, 2012.

[40] D. N. Lee. A theory of visual control of braking based on information about time to collision. *Perception*, 5:437–59, 1976.

[41] David Lee. Guiding movements by coupling taus. *Ecological Psychology*, 10:221–250, 1998.

[42] David N Lee, Mark NO Davies, Patrick R Green, et al. Visual control of velocity of approach by pigeons when landing. *Journal of Experimental Biology*, 180(1):85–104, 1993.

[43] David N Lee, Paul E Reddish, and DT Rand. Aerial docking by hummingbirds. *Naturwissenschaften*, 78(11):526–527, 1991.

[44] David N Lee, J Roly Lishman, and James A Thomson. Regulation of gait in long jumping. *Journal of Experimental Psychology: Human perception and performance*, 8(3):448, 1982.

[45] Gareth Padfield, David Lee, and Roy Bradley. How do helicopter pilots know when to stop, turn or pull up? *Journal of the American Helicopter Society*, 48(2):108–119, April 2003.

[46] Gareth D Padfield and A Taghizad. How long do pilots look forward? In *31st European Rotorcraft Forum, Florence, Italy*, 2005.

[47] GD Padfield. The tau of flight control. *Aeronautical Journal*, 115(1171):521–556, 2011.

[48] M. Srinivasan, S. Zhang, and J. Reinhard. Small brains, smart minds: vision, perception, navigation and'cognition'in insects. *Invertebrate vision. Cambridge University Press, Cambridge*, pages 462–493, 2006.

[49] Floris van Breugel and Michael H. Dickinson. The visual control of landing and obstacle avoidance in the fruit fly drosophila melanogaster. *The Journal of Experimental Biology*, 2012.

[50] Ali Borji and Laurent Itti. State-of-the-art in visual attention modeling. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):185–207, 2013.

[51] Laurent Itti and Christof Koch. Computational modelling of visual attention. *Nature reviews neuroscience*, 2(3):194–203, 2001.

[52] Mohammad Lone and Alastair Cook. Review of pilot modeling techniques. In *AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, 2010.

[53] A. Tustin. The nature of the operator's response in manual control, and its implications for controller design. *Electrical Engineers - Part IIA: Automatic Regulators and Servo Mechanisms, Journal of the Institution of*, 94(2):190 –206, May 1947.

[54] Lindsay Russell. Characteristics of the Human as a Linear Servo-Element. Master's thesis, MIT, Cambridge, Mass., 1951.

[55] Jerry Elkind. *Characteristics of simple manual control systems*. PhD thesis, MIT, Cambridge, Mass., 1956.

[56] Duane McRuer and Henry Jex. A Review of Quasi-Linear Pilot Models. *IEEE Transactions on Human Factors in Electronics*, HFE-8:231–249, 1967.

[57] Richard J. Jagacinski and John M. Flach. *Control Theory for Humans, Quantitative Approaches to Modeling Performance*. Lawrence Erlbaum Associates, New Jersey, 2003.

[58] Duane McRuer and Ezra Kendel. Mathematical models of human pilot behavior. Technical Report AGARDograph 180, Advisory Group on Aerospace Research and Development, January 1974.

[59] Duane McRuer. Human dynamics in man-machine systems. *Automatica*, 16(3):237–253, 1980.

[60] D.L. Kleinman, S. Baron, and W.H. Levison. An optimal control model of human response part i: Theory and validation. *Automatica*, 6(3):357 – 369, 1970.

[61] Ronald A. Hess. Structural model of the adaptive human pilot. *Journal of Guidance, Control, and Dynamics*, 3(5):416–423, September 1980.

[62] Ronald A Hess. A rationale for human operator pulsive control behavior. *Journal of Guidance, Control, and Dynamics*, 2(3):221–227, 1979.

[63] R. A. Hess and K. K. Chan. Preview control pilot model for near-earth maneuvering helicopter flight. *Journal of Guidance, Control, and Dynamics*, 11(2):146–152, March 1988.

[64] Ronald A Hess. Pursuit tracking and higher levels of skill development in the human pilot. *Systems, Man and Cybernetics, IEEE Transactions on*, 11(4):262–273, April 1981.

[65] RA Hess and A Modjtahedzadeh. A control theoretic model of driver steering behavior. *IEEE Control Systems Magazine*, 10(5):3–8, 1990.

[66] Ezra S Krendel and Duane T McRuer. A servomechanisms approach to skill development. *Journal of the Franklin Institute*, 269(1):24–42, 1960.

[67] R.K. Heffley. "A Compilation and Analysis of Helicopter Handling Qualities Data; Volume II: Data Analysis". Technical Report CR-3145, NASA, 1979.

[68] Ronald Hess and B. McNally. Automation effects in a multiloop manual control system. *IEEE Transactions on Systems, Man, and Cybernetics*, 16(1):111–121, January 1986.

[69] Mark White, Gareth Padfield, and Robert Armstrong. Progress in measuring simulation fidelity using an adaptive pilot model. In *Proceedings of the 60th Annual Forum of the American Helicopter Society*. American Helicopter Society, 2004.

[70] Robert K Heffley. A pilot-in-the-loop analysis of several kinds of helicopter acceleration/deceleration maneuvers. *NASA CP*, 2216, 1982.

[71] Michael F Land. Eye movements and the control of actions in everyday life. *Progress in retinal and eye research*, 25(3):296–324, 2006.

[72] Mary Hayhoe and Dana Ballard. Eye movements in natural behavior. *Trends in cognitive sciences*, 9(4):188–194, 2005.

[73] Leif Johnson, Brian Sullivan, Mary Hayhoe, and Dana Ballard. Predicting human visuomotor behaviour in a driving task. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 369(1636):20130044, 2014.

[74] D Guitton and M Volle. Gaze control in humans: eye-head coordination during orienting movements to targets within and beyond the oculomotor range. *Journal of neurophysiology*, 58(3):427–459, 1987.

[75] Emilio Bizzi. Eye-head coordination. *Comprehensive Physiology*, 1981.

[76] Indika Wijayasinghe, Eugenio Aulisa, and BK Ghosh. Tracking and Optimal Control Problems in Human Head/Eye Coordination. In *American Control Conference*, 2013.

[77] K-S Chun and DA Robinson. A model of quick phase generation in the vestibuloocular reflex. *Biological cybernetics*, 28(4):209–221, 1978.

[78] Laurence R Young and David Sheena. Survey of eye movement recording methods. *Behavior research methods & instrumentation*, 7(5):397–429, 1975.

[79] Bérénice Mettler, Zhaodan Kong, Bin Li, and Jonathan Andersh. Systems view on spatial planning and perception based on invariants in agent-environment dynamics. *Frontiers in neuroscience*, 8, 2014.

[80] Dario D Salvucci and Joseph H Goldberg. Identifying fixations and saccades in eye-tracking protocols. In *Proceedings of the 2000 symposium on Eye tracking research & applications*, pages 71–78. ACM, 2000.

[81] Oleg V Komogortsev, Sampath Jayarathna, Do Hyong Koh, and Sandeep Munikrishne Gowda. Qualitative and quantitative scoring and evaluation of the eye movement classification algorithms. In *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*, pages 65–68. ACM, 2010.

[82] Frouke Hermens, Rhona Flin, and Irfan Ahmed. Eye movements in surgery: A literature review. *Journal of Eye Movement Research*, 6(4), 2013.

[83] Susana Martinez-Conde, Stephen L Macknik, and David H Hubel. The role of fixational eye movements in visual perception. *Nature Reviews Neuroscience*, 5(3):229–240, 2004.

[84] B. Mettler, N. Dadkhah, Z. Kong, and J. Andersh. Infrastructure and research framework for human interactive and autonomous guidance. In *International Conference on Unmanned Aircraft Systems (ICUAS'12)*, 2012.

[85] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng. Ros: an open-source robot operating system. In *ICRA Workshop on Open Source Software*, 2009.

[86] Vicon. *Vicon MX Systems*, October 2006.

[87] Navid Dadkhah and Bernie Mettler. System identification modeling and flight characteristics analysisof miniature co-axial helic. *Journal of the American Helicopter Society*, 2012.

[88] SensoMotoric Instruments GmbH, Warthestraße 21 , D-14513 Teltow/Berlin , Germany. *iView ETG User Manual*, version 1.1 edition, April 2012.

[89] K. M. Wurm, A. Hornung, M. Bennewitz, C. Stachniss, and W. Burgard. OctoMap: A probabilistic, flexible, and compact 3D map representation for robotic systems. In *Proc. of the ICRA 2010 Workshop on Best Practice in 3D Perception and Modeling for Mobile Manipulation*, Anchorage, AK, USA, May 2010.

[90] Jon Andersh, Bernard Mettler, and Nikos Papanikolopoulos. Miniature embedded rotorcraft platform for aerial teleoperation experiments. In *Proceedings of the 2009 17th Mediterranean Conference on Control and Automation*, Thessaloniki, Greece, 2009. 17th Mediterranean Conference on Control and Automation.

[91] Jonathan Andersh and Bernie Mettler. System integration of a miniature rotorcraft for aerial tele-operation research. *Mechatronics*, 21(5):776–788, 2011.

[92] E-flite Advancing Electric Flight. `http://www.e-fliterc.com`.

[93] M. B. Tischler and R. K. Remple. *Aircraft and Rotorcraft System Identification, Engineering Methods with Flight Test Examples*. AIAA Education Series, 2006.

[94] US Army Aeroflightdynamics Directorate. *CIFER Comprehensive Identification from Frequency Responses User's Guide*, 2010.

[95] Bernard Mettler. *Identification modeling and characteristics of miniature rotorcraft*. Kluwer Academic Publisher, 2003.

[96] G. D. Padfield. *Helicopter Flight Dynamics: The Theory and and Application of Flying Qualities and Simulation Modeling*. AIAA Education Series, 1996.

[97] Christopher D Wickens, Justin G Hollands, Simon Banbury, and Raja Parasuraman. *Engineering psychology & human performance*. Psychology Press, 2015.

[98] Bérénice Mettler, Navid Dadkhah, Zhaodan Kong, and Jonathan Andersh. Research infrastructure for interactive human-and autonomous guidance. *Journal of Intelligent & Robotic Systems*, 70(1-4):437–459, 2013.

[99] Paul M Fitts. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of experimental psychology*, 47(6):381, 1954.

[100] Richard A Schmidt, Howard Zelaznik, Brian Hawkins, James S Frank, and John T Quinn Jr. Motor-output variability: a theory for the accuracy of rapid motor acts. *Psychological review*, 86(5):415, 1979.

[101] U.S. Army Aviation and Missile Command, Aviation Engineering Directorate. *Handling Qualities Requirements for Military Rotorcraftt, ADS-33E-PRF*, March 2000.

[102] R. Brockett. Minimum attention control. In *Proceedings of the 36th Conference on Decision & Control*, December 1997.

[103] Adolph Atencio. Fidelity assessment of a uh-60a simulation on the nasa ames vertical motion simulator. Technical report, NASA Langley Research Center, 1993.

[104] Frank M Drop, Daan M Pool, Herman J Damveld, Marinus M van Paassen, and Max Mulder. Identification of the feedforward component in manual control with predictable target signals. *IEEE transactions on cybernetics*, 43(6):1936–1949, 2013.

[105] DM Pool, MM Van Paassen, and Max Mulder. *Modeling human dynamics in combined ramp-following and disturbance-rejection tasks*. American Institute of Aeronautics and Astronautics (AIAA), 2010.

[106] Bérénice Mettler and Jon Andersh. Fundamental control characteristics of curvilinear motion in human and automatic path tracking tasks. In *2013 American Control Conference*, pages 6460–6467. IEEE, 2013.

[107] E. Todorov. Optimality principles in sensorimotor control. *Nature neuroscience*, 7(9):907–915, 2004.

[108] M.B. Tischler and M.G. Cauffman. *"Comprehensive Identification from Frequency Responses; An interactive facility for system identification and verification, Class Notes and User's Manual"*. NASA Ames Research Center, Moffett Field, CA, 1994.

[109] Karl LaFleur, Kaitlin Cassady, Alexander Doud, Kaleb Shades, Eitan Rogin, and Bin He. Quadcopter control in three-dimensional space using a noninvasive motor imagery-based brain–computer interface. *Journal of neural engineering*, 10(4):046003, 2013.

[110] Stephen J Huston and Holger G Krapp. Visuomotor transformation in the fly gaze stabilization system. *PLoS Biol*, 6(7):e173, 2008.

[111] DA Robinson. The mechanics of human saccadic eye movement. *The Journal of physiology*, 174(2):245–264, 1964.

[112] DA Robinson. The mechanics of human smooth pursuit eye movement. *The Journal of Physiology*, 180(3):569, 1965.

[113] D Boghen, BT Troost, and RB Daroff. Velocity characteristics of normal human saccades. *Investigative Ophthalmology*, 13(8):619–623, 1974.

[114] Robert Yee, Edward Wong, Robert Baloh, and Vicente Honrubio. A study of congenital nystagmus waveforms. *Neurology*, 26(4):326–326, 1976.

[115] John Brown. Some tests of the decay theory of immediate memory. *Quarterly Journal of Experimental Psychology*, 10(1):12–21, 1958.

[116] Martin Voshell, David D Woods, and Flip Phillips. Overcoming the keyhole in human-robot coordination: simulation and evaluation. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 49, pages 442–446. Sage Publications, 2005.

[117] CC Smyth, V Paul, AM Meldrum, and K McDowell. Examining alternative display configurations for an indirect vision driving interface. *US Army Research Laboratory: Aberdeen Proving Ground, MD, to be published*.

[118] Christopher C Smyth. Indirect vision driving with fixed flat panel displays for near unity, wide, and extended fields of camera view. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 44, pages 541–544. SAGE Publications, 2000.

[119] Dingyun Zhu, Tom Gedeon, and Ken Taylor. moving to the centre: A gaze-driven remote camera control for teleoperation. *Interacting with Computers*, 23(1):85–95, 2011.

[120] Hemin Omer Latif, Nasser Sherkat, and Ahmad Lotfi. Remote control of mobile robots through human eye gaze: The design and evaluation of an interface. In *SPIE Europe Security and Defence*, pages 71120X–71120X. International Society for Optics and Photonics, 2008.

[121] Eliseo Stefano Maini, Giancarlo Teti, Michele Rubino, Cecilia Laschi, and Paolo Dario. Bio-inspired control of eye-head coordination in a robotic anthropomorphic head. In *Biomedical Robotics and Biomechatronics, 2006. BioRob 2006. The First IEEE/RAS-EMBS International Conference on*, pages 549–554. IEEE, 2006.

[122] Tak Kit Lau, Yun hui Liu, and Kai wun Lin. A robust state estimation method against gnss outage for unmanned miniature helicopters. In *IEEE International Conference on Robotics and Automation*, pages 1116–1122, may 2010.

[123] Myungsoo Jun, S.I. Roumeliotis, and G.S. Sukhatme. State estimation of an autonomous helicopter using kalman filtering. In *International Conference on Intelligent Robots and Systems*, volume 3, pages 1346–1353, 1999.

[124] Robert Mahony and Tarek Hamel. *Advances in Unmanned Aerial Vehicles: State of the Art and the Road to Autonomy*, chapter Robust Nonlinear Observers for Attitude Estimation of Mini UAVs, pages 343–375. Springer, 2007.

[125] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004.

[126] H. Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, pages 133 – 135, 1981.

[127] D.J. Heeger and A.D. Jepson. Subspace methods for recovering rigid motion I: Algorithm and implementation. *International Journal of Computer Vision*, 7(2):95–117, 1992.

[128] S. Soatto, R. Frezza, and P. Perona. Motion estimation via dynamic vision. *IEEE Transactions on Automatic Control*, 41(3), 1996.

[129] J. Kehoe, A. Watkins, R. Causey, and R. Lind. State estimation using optical flow from parallax-weighted feature tracking. *Proc. of the 2006 AIAA Guidance, Navigation and Control Conference*, August 2006.

[130] S. Soatto and P. Perona. Recursive 3-d motion estimation using subspace constraints. *International Journal of Computer Vision*, 22(3), 1997.

[131] Adam J Rutkowski, Roger D Quinn, and Mark A Willis. Biologically inspired self-motion estimation using the fusion of airspeed and optical flow. In *American Control Conference*, pages 6–pp. IEEE, 2006.

[132] Dave Zachariah and Magnus Jansson. Self-motion and wind velocity estimation for small-scale uavs. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1166–1171, may 2011.

[133] R. Madison, G. Andrews, and P. DeBitetto. Vision-aided navigation for small uavs in gps-challenged environment. *Proc. of the 2007 AIAA Guidance, Navigation and Control Conference*, 2007.

[134] Shaojie Shen, Yash Mulgaonkar, Nathan Michael, and Vijay Kumar. Vision-based state estimation for autonomous rotorcraft mavs in complex environments. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 1758–1764. IEEE, 2013.

[135] Markus Achtelik, Stephan Weiss, and R Siegwart. Onboard IMU and monocular vision based control for MAVs in unknown in and outdoor environments. In *IEEE international conference on Robotics and automation*, pages 3056–3063. IEEE, 2011.

[136] Gabriele Bleser and Gustaf Hendeby. Using optical flow for filling the gaps in visual-inertial tracking. In *European Signal Processing Conference*, 2010.

[137] Stephan Weiss, Markus W Achtelik, Simon Lynen, Margarita Chli, and Roland Siegwart. Real-time onboard visual-inertial state estimation and self-calibration of mavs in unknown environments. In *IEEE International Conference on Robotics and Automation*, pages 957–964. IEEE, 2012.

[138] A. Johnson and L. Matthies. Precise image-based motion estimation for autonomous small body exploration. *Artificial Intelligence, Robotics and Autonomation in Space (iSAIRAS)*, pages 627–634, 1999.

[139] H. Rotstein and P. Gurfil. Aircraft state estimation from visual motion: application of the subspace constraints approach. In *Proceedings of IEEE Position Location and Navigation Symposium*, pages 263–270, 2002.

[140] T. Webb, R. Prazenica, A. Kurdila, and R. Lind. Vision-based state estimation for autonomous micro air vehicles. *AIAA Guidance, Navigation, and Control Conference*, 2004.

[141] Pierre-Jean Bristeau, Francois Callou, David Vissiere, and Nicolas Petit. The navigation and control technology inside the AR drone micro UAV. In *18th IFAC World Congress*, pages 1477–1484, 2011.

[142] A.-J. Baerveldt and R. Klang. A low-cost and low-weight attitude estimation system for an autonomous helicopter. In *IEEE International Conference on Intelligent Engineering Systems*, pages 391–395, 1997.

[143] Pierre-Jean Bristeau, Philippe Martin, Erwan Salaun, and Nicolas Petit. The Role of Propeller Aerodynamics in the Model of a Quadrotor UAV. In *Proceedings of the European Control Conference 2009*, 2009.

[144] M. Calonder, V. Lepetit, C. Strecha, and P. Fua. BRIEF: Binary robust independent elementary features. *European Conference on Computer Vision*, pages 778–792, 2010.

[145] S.S. Beauchemin and J.L. Barron. The computation of optical flow. *ACM Computing Surveys (CSUR)*, 27(3):433–466, 1995.

[146] K. Kanatani. *Statistical optimization for geometric computation: theory and practice*. Dover Publications Incorporated, 2005.

[147] Y. Ma, S. Soatto, J. Kosecka, and S.S. Sastry. *An Invitation to 3-D Vision*. Springer-Verlag, LCC, 2003.

[148] W. Niehsen. Information fusion based on fast covariance intersection filtering. In *Proceedings of the Fifth International Conference on Information Fusion*, volume 2, pages 901–904. IEEE, 2002.